



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

DESARROLLO DE UN SISTEMA DE
BASE DE DATOS PARA EL CENTRO
DE CALCULO DE LA FACULTAD DE
INGENIERIA

T E S I S

QUE PARA OBTENER EL TITULO DE
INGENIERO EN COMPUTACION

P R E S E N T A N

GERARDO CABALLERO ESPEJO
JULIA PALOMA ESTRADA MOLINA
NESTOR GOMEZ MUÑOZ
GABRIELA SOTO LOPEZ

DIRECTOR DE TESIS

ING. SOCRATES A. MUÑOZ ZAFRA



FALL DE INGEN

MEXICO, D. F.

AGOSTO DE 1990



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

Contenido

Introducción	1
Capítulo 1 Antecedentes	5
1.1 Bases de Datos	5
1.1.1 Definición y propósitos	6
1.1.2 Modelos lógicos	11
1.1.2.1 Modelos lógicos basados en registros	14
1.1.2.2 Modelos lógicos basados en objetos	22
1.1.3 Modelos y Esquemas	23
1.1.4 Manejadores de Bases de Datos	25
1.1.4.1 Lenguajes	27
1.1.4.2 El Administrador de la Base de Datos	28
1.1.4.3 Usuarios de la Base de Datos	29
1.1.4.4 Estructura del Sistema	30
1.2 Modelado de la Base de Datos	30
1.2.1 Fronteras	32
1.2.1.1 Alcance de integración	32
1.2.1.2 Dimensiones	33

1.2.1.3	Vistas de datos	33
1.2.2	Búsqueda de requisitos	34
1.2.2.1	Identificación de líderes	35
1.2.2.2	Entrevistas	36
1.2.2.3	Cuestionarios	37
1.2.2.4	Observación	37
1.2.3	Entidades y Asociaciones	38
1.2.4	Esqueletos y tablas normalizadas	39
1.2.5	Definiciones	40

Capítulo 2 Modelado

43

2.1	Fronteras	43
2.2	Búsqueda de requisitos	46
2.3	Entidades y Asociaciones	47
2.4	Esqueletos y tablas normalizadas	51
2.4.1	Esqueletos	51
2.4.2	Tablas y atributos	53
2.5	Definiciones	63
2.5.1	Definición de Entidades	64
2.5.2	Definición de Asociaciones	67
2.5.3	Definición de Atributos	70
2.5.4	Tablas ejemplares	91
2.5.5	Tablas cruzadas	118
2.5.6	Tabla de accesos	121

Capítulo 3 Selección de un Manejador de Bases de Datos

129

3.1	Alternativas	129
3.2	Estudio teórico	130
3.2.1	Características generales del SMBD	131

3.2.1.1	Antecedentes	131
3.2.1.2	Administración del espacio en discos	132
3.2.1.3	Representación y definición de datos	133
3.2.1.4	Métodos de acceso y recuperación	144
3.2.1.5	Control de acceso múltiple	147
3.2.1.6	Límites del SMBD	150
3.2.2	Características y facilidades para el usuario	151
3.2.2.1	Facilidades de consulta	151
3.2.2.2	Facilidades de actualización	153
3.2.2.3	Diccionario de datos	153
3.2.2.4	Interacción con otros lenguajes	157
3.2.2.5	Productos adicionales	158
3.2.3	Características de administración	161
3.2.3.1	Seguridad	161
3.2.3.2	Respaldo y recuperación de errores	164
3.2.3.3	Facilidades para monitorear	168
3.2.3.4	Utilerías para administración en general	170
3.2.4	Facilidades que ofrece el proveedor	173
3.2.4.1	Soporte	173
3.2.4.2	Calidad de la documentación disponible	173
3.2.5	Requerimientos del SMBD	174
3.2.5.1	Cantidad de espacio en disco	174
3.2.5.2	Cantidad de memoria principal	176
3.3	Pruebas experimentales	177
3.4	Conclusiones	190

5.5	Diseño y Desarrollo del sistema	240
Capítulo 6	Conclusiones y comentarios finales	245
6.1	Comentarios sobre el modelo	245
6.2	Comentarios sobre la selección	247
6.3	Comentarios sobre la implantación	247
6.4	Comentarios sobre la aplicación	249
Apéndice A	El Modelo de Entidades y Asociaciones	252
A.1	Conceptos básicos	252
A.2	Diagramas de Entidades y Asociaciones	254
A.3	Mapeo de UER a esqueletos y tablas	261
Apéndice B	Normalización	270
Bibliografía		274

Introducción

Desde sus orígenes, el Centro de Cálculo ha venido manejando información de importancia para la Facultad de Ingeniería, información que ha incrementado su volumen y la cantidad de aplicaciones realizadas sobre ella. Sin embargo, las técnicas para organizar tal información y las formas de utilizarla, continúan siendo semejantes a las aplicadas en un principio.

Se han cambiado computadoras, medios de almacenamiento, herramientas de "software", y otras cosas más, pero se continúa desarrollando con los conceptos de "programa de aplicación, archivos de entrada y archivos de salida", lo cuál, entre una gran diversidad de intereses y puntos de vista, crea un ambiente complicado y lleno de redundancias e inconsistencias que representan costos de tiempo y esfuerzo en mantenimiento improductivo, que podrían ser reducidos.

La técnica de Bases de Datos es una alternativa que puede ayudar a organizar más eficientemente los recursos de información del Centro, pero que no ha sido utilizada por la carencia de un adecuado Sistema Manejador de Bases de Datos. Cuando fueron donados a la Facultad de Ingeniería dos de los manejado-

res que hay en el mercado de cómputo para los sistemas VAX, fue posible iniciar el cambio hacia las Bases de Datos y los lenguajes de cuarta generación.

Este cambio no es sencillo. Si no se fundamenta la nueva forma de trabajo en un firme y completo análisis antes de su aplicación, se podría caer en los mismos problemas que actualmente se enfrentan. Es por eso que las partes modulares de este trabajo se refieren a la etapa del modelado de datos y a la definición del esquema conceptual, puesto que cualquier aplicación posterior deberá ser satisfecha a partir del modelo obtenido.

El objetivo primordial de este trabajo es el de modelar una Base de Datos capaz de satisfacer todas las demandas de información del Centro de Cálculo, y por lo tanto de la Facultad de Ingeniería, e implantarla en el Sistema Manejador de Bases de Datos que demuestre ser superior. No se pretende dejar funcionando todos los sistemas que a la fecha existen en el CECAFI, sino establecer las bases suficientes para un posterior mecanismo de migración al nuevo ambiente, con la participación de todos los departamentos que conforman al Centro.

Sin embargo, como culminación del trabajo es imprescindible desarrollar una aplicación que compruebe que el Modelo funciona, es aplicable y digno de confianza, para lo que Atención Personal (Inscripciones en Línea) es un sistema ideal.

Adicionalmente, con este reporte escrito, se pretende cumplir con tres objetivos: describir el análisis y modelado que se requiere para definir el esquema; dejar establecida

una metodología de diseño de bases de datos, que pueda servir como un fundamento didáctico y tener un documento que pueda ser consultado por quienes utilicen en un futuro la Base de Datos del CEGAFI.

Se presentan siete capítulos y una sección de apéndices, como se describe a continuación:

En el capítulo uno se define el término Base de Datos y los conceptos y terminología utilizada en tal ambiente; además se describen los pasos utilizados en el proceso de modelado.

En el capítulo dos se presentan los productos obtenidos durante el proceso de modelado: se definen las fronteras y se explica cómo fueron determinados los requisitos por cumplir; se presenta el Diagrama de Entidades y Asociaciones, junto con los Esqueletos y Tablas obtenidas; al final se enlistan las definiciones de Entidades, Asociaciones y Atributos, y se muestran las tablas ejemplares.

El capítulo tres describe los pasos y criterios seguidos para la selección del Manejador de Bases de Datos, desde el estudio de las características de los candidatos, hasta la determinación del más adecuado para los intereses del Centro; también se muestran tablas comparativas y los programas utilizados para hacer las pruebas de respuesta del sistema.

En el capítulo cuatro se narra la forma en que fue implantado el Esquema Conceptual con base en el Modelo y cómo se determinó la organización particular de manejador y de sus archivos; además se describen los pasos a seguir para la migración de la información al nuevo ambiente.

En el capítulo cinco se presentan los antecedentes y procesos desarrollados para la obtención del Sistema de Atención Personal, como primera aplicación dentro del ambiente de Bases de Datos.

Finalmente, en el capítulo seis se resumen las principales observaciones y conclusiones obtenidas al desarrollar este trabajo. Tales conclusiones son agrupadas de acuerdo a la etapa en que fueron encontradas durante el desarrollo.

En la sección de apéndices se describe la forma de utilizar el Modelo de Entidades y Asociaciones, así como las reglas de normalización, ambos conceptos empleados en el momento de hacer el modelo lógico de la Base de Datos.

Antecedentes

El diseñar e implantar una Base de Datos requiere de conocer, tanto conceptos generales de bases de datos, como los elementos que se deben considerar al hacer el modelado de los datos.

En este capítulo se presentan los tópicos mínimos que toda persona que trabaja con una base de datos debe conocer, así como la descripción de los pasos que se siguen en el modelado de datos. Tanto de conceptos generales como del proceso de modelado existen temas que aquí no se tratan, además de que para diversos grupos de desarrollo y autores de textos existen discrepancias en sus puntos de vista. Sin embargo, consideramos que este enfoque es completo para los objetivos que se desean cumplir.

1.1 Bases de Datos

El concepto de Base de Datos surgió hace ya mucho tiempo, aunque sus definiciones han ido evolucionando al adaptarse al punto de vista de cada tendencia o enfoque que se le dio. Para tratar de homogeneizar las ideas, se han desarrollado modelos

de abstracción que han sido muy aceptados y sobre los que se han hecho importantes trabajos.

A continuación se verá qué es, para qué sirve, y de qué se compone una Base de Datos.

1.1.1 Definición y propósitos

La definición de Base de Datos es paradójicamente tan amplia y concisa que, más que una, se presentan varias definiciones:

C. J. Date (An Introduction to Database Systems), "Una Base de Datos es una colección de datos de operación almacenados y utilizados por los sistemas de aplicación de una empresa específica."

IBM Data Processing Division (Data Processing Glossary), "conjunto de datos, parte del total de otro conjunto de datos, que es suficiente para un propósito dado o para un sistema de procesamiento de datos dado."

Ronald Ross (Data Management Systems), "Una colección de elementos de datos o registros únicos interrelacionados, en uno o más archivos de computadora, los cuales pueden ser procesados por múltiples programas de aplicación."

Leo Cohen (Data Base Management Systems), "(a) Los elementos de datos agregados que comprenden el conjunto de archivos y registros de un sistema dado o de un conjunto de sistemas. (b) Una colección de archivos que tienen interrelaciones lógicas."

(c) Una colección no redundante de elementos de datos interrelacionados por una o más aplicaciones."

Alan Mayne (Introducing Relational Database), "Una Base de Datos es una colección de datos almacenados, organizados en tal forma que todos los requisitos de datos de los usuarios son satisfechos por esa Base de Datos. En general existe sólo una copia lógica de cada elemento de datos."

Roger S. Pressman (Software Engineering: A practitioner's Approach) "Colección de información organizada en tal forma que facilite el acceso, el análisis y la generación de reportes."

Y para nosotros la más completa:

James Martin (Principles of Data-Base Management y Computer Data-Base Organization), "Una colección de datos interrelacionados, almacenados juntos con redundancia controlada para servir a una o más aplicaciones; los datos son almacenados de esta manera para que sean independientes de los programas que los usan; se emplea un enfoque común y controlado en la adición de nuevos datos, y en la modificación y recuperación de los datos ya existentes en la Base de Datos."

De las definiciones anteriores es posible extraer ocho conceptos fundamentales:

Sistemas de aplicación de la Empresa¹. Puesto que es la Empresa lo más importante, y lo que ha creado la necesidad de una Base de Datos, esta última debe servir para satisfacer a la Empresa a través de tales sistemas.

Colección de datos de operación. La Base de Datos se compone de una colección de datos que representan a la Empresa y que serán empleados para la operación de sus sistemas.

Datos interrelacionados. Para que los datos correspondan a una misma Base de Datos, debe existir una relación lógica entre ellos.

No redundancia. Uno de los problemas que se intenta solucionar con una Base de Datos es el de la repetición indiscriminada de datos. Esa es la redundancia.

Archivos de computadora almacenados juntos. Se ha asumido que la información será mantenida en sistemas de cómputo, pero lo importante es que toda una Base de Datos se encuentre en el mismo sistema, ya sea en el mismo disco, bajo el mismo sistema operativo, o dentro de la misma red.

Facilidad de acceso, análisis y producción. Se deben tener o proporcionar los mecanismos necesarios para que el acceso a la información sea lo más simple posible, no sólo evitando crear barreras artificiales entre los datos y los usuarios, sino dando las herramientas necesarias para su rápida explotación.

¹ En adelante se empleará la palabra Empresa para hacer referencia a la organización o entidad que requiere de la Base de Datos. La empresa puede ser un emporio comercial, una escuela, un departamento de gobierno, etc., todo depende de los límites que sean fijados.

Independencia programa-datos. Cuanto más independientes de los datos sean los programas de aplicación, y viceversa, más inmunes serán los sistemas a la evolución natural que sufre cualquier base de Datos.

Enfoque común. El hecho de que las operaciones realizadas sobre la Base de Datos sean controladas por un mismo sistema, inclusive con un sólo administrador para su custodia, evita los tan caóticos problemas de comunicación, y disminuye los efectos de las normales diferencias de opinión entre las secciones de la Empresa.

El sistema que se menciona anteriormente es conocido como Sistema Manejador de la Base de Datos (SMRD), este sistema es el encargado de hacer la traducción entre la perspectiva global de los datos en la base de datos y la perspectiva local esperada por cada programa de aplicación. El propósito fundamental de tal sistema es que, dada una Base de Datos, proporcione un medio ambiente conveniente y eficiente para su utilización.

El término Manejador de Base de Datos contrasta con el de Manejador de Archivos. Este último se emplea en los sistemas desarrollados en forma tradicional. Podemos observar sus características a continuación:

Manejador de archivos

Redundancia e inconsistencia de datos.

Dificultad para tener acceso a los datos.

Aislamiento de datos.

Usuarios múltiples sin control.

Problemas de seguridad

Manejador de BD

Interacción con el administrador de archivos.

Integridad.

Seguridad.

Respaldo y recuperación.

Control de concurrencia.

Herramientas para la explotación.

El Sistema Manejador de Bases de Datos intenta solucionar los problemas generados por los Manejadores de archivos. La redundancia e inconsistencia de datos son atacados con un adecuado diseño de datos desde una perspectiva global de la Empresa y colocando a todos esos datos juntos bajo el mismo sistema. La dificultad para tener acceso a los datos y el aislamiento son evitados al proporcionar un lenguaje y herramientas que permitan obtener la información requerida, relacionando todos los datos de la Base. Finalmente, los problemas de seguridad los soluciona el Manejador fijando restricciones y privilegios entre los datos y los usuarios.

Otro aspecto importante que debe ser atacado por un SMDB es el mantenimiento improductivo, esto se refiere a la inversión de tiempo y esfuerzo en la corrección de programas de aplicación que no mejorarán sus funciones. Esto último se da cuando, por ejemplo, se agrega un campo más a los archivos de datos, obligando a la reprogramación de todos aquellos sistemas que utilizan al archivo afectado.

A continuación se muestran las figuras 1.1 y 1.2, en donde se aprecia cómo el modificar la vista global de la Base de Datos es transparente a casi todos los programas de aplicación.

Posteriormente se verán más características de los Sistemas Manejadores de Bases de Datos (§1.1.4).

1.1.2 Modelos lógicos

Se ha visto que una base de datos se compone esencialmente de datos, que estos se encuentran en los medios de almacenamiento de una computadora, que existen mecanismos para tener un acceso rápido a los mismos, etc., pero cuando vamos a definir qué datos estarán guardados y cuál será la organización que tendrán, pensar al nivel de bloques de disco, registros, apuntadores y segmentos de memoria sería demasiado complejo para nuestra limitada mente humana.

Es por lo anterior que, tratando de encontrar una representación más accesible, se han desarrollado modelos que ilustran la lógica del comportamiento, las restricciones, la semántica y las relaciones que existen entre los datos, en una forma fácil de entender, organizar y modificar. Entre esos modelos, de los cuales hay una variedad, los más conocidos son los modelos lógicos basados en registros y los modelos lógicos basados en objetos.

programa cambios_producto

No. prod.	descripcion	precio	cantidad_en_stock	cantidad_de_reorden
-----------	-------------	--------	-------------------	---------------------

programa reporte_reorden

No. prod.	descripcion	cantidad_en_stock	cantidad_de_reorden
-----------	-------------	-------------------	---------------------

programa cambio_precios

No. prod.	precio
-----------	--------

programa impresion_catalogo

No. prod.	descripcion	precio
-----------	-------------	--------

programa movimientos_stock

No. prod.	cantidad_en_stock
-----------	-------------------

Manejador
de
Base de Datos

No. prod.	descripcion	precio	cantidad_en_stock	cantidad_de_reorden
-----------	-------------	--------	-------------------	---------------------

Figura 1.1 Base de Datos y aplicaciones antes de la modificación.

programa cambios_producto

No. prod.	descripcion	precio	cantidad_en_stock	cantidad_de_reorden	minimo_stock
-----------	-------------	--------	-------------------	---------------------	--------------

programa reporte_reorden

No. prod.	descripcion	cantidad_en_stock	cantidad_de_reorden	minimo_stock
-----------	-------------	-------------------	---------------------	--------------

programa cambio_precios

No. prod.	precio
-----------	--------

programa impresion_catalogo

No. prod.	descripcion	precio
-----------	-------------	--------

programa movimientos_stock

No. prod.	cantidad_en_stock
-----------	-------------------

Manejador
de
Base de Datos

No. prod.	descripcion	precio	cantidad_en_stock	cantidad_de_reorden	minimo_stock
-----------	-------------	--------	-------------------	---------------------	--------------

Figura 1.2 Base de Datos y aplicaciones después de la modificación.

1.1.2.1 Modelos lógicos basados en registros

Han sido los más utilizados durante mucho tiempo. Estos modelos representan a la realidad tomando como base la forma en que los datos son almacenados en la computadora y cómo son mantenidas las asociaciones entre ellos. El principal inconveniente que se puede observar en ellos es el hecho de que al modelar con estas herramientas, se crea una dependencia muy grande de tipo de organización física que se emplea para llevar el esquema de la base de datos. En este grupo se encuentran los siguientes modelos:

Modelo Jerárquico. En este modelo, los datos se representan como una colección de registros, mientras que sus vínculos son ligas o apuntadores entre ellos. Los registros contienen campos de información relativa, mientras que las ligas representan los tipos de asociaciones que pueden existir entre esos registros. Esto se ejemplifica a continuación en la Figura 1.3:

Se puede observar que la representación es de tipo arborescente, y que un registro solo se encuentra vinculado con otro de mayor jerarquía, mientras que a su vez puede tener ligados a varios de jerarquía inferior. De lo anterior se desprende que las asociaciones que puede representar este modelo son 1:1 y 1:N, esto es, un registro ligado con otro, y un registro ligado con varios.

El auxilio gráfico que se emplea al modelar este tipo de representación se conoce como **diagramas de estructura de árbol**

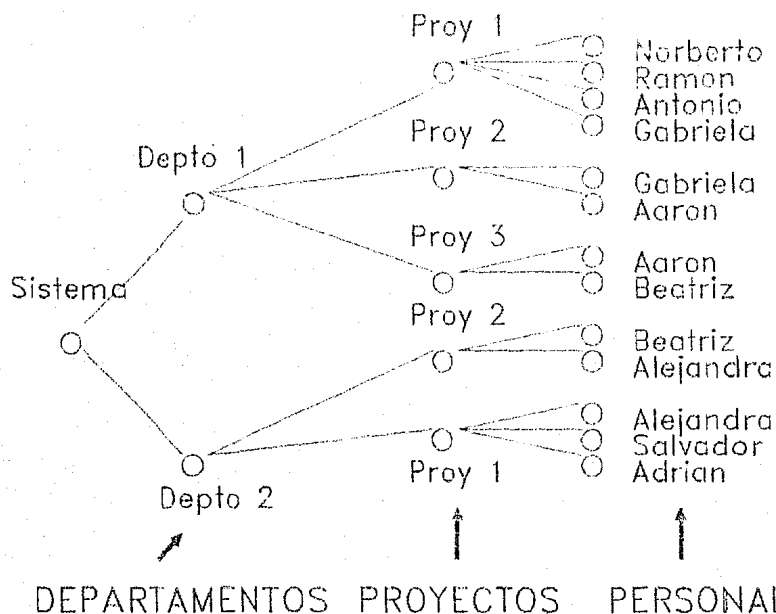


Figura 1.3 Modelo Jerárquico

(figura 1.4). Los rectángulos representan tipos de registro, mientras que las flechas representan ligas entre ellos. La punta de la flecha señala al registro de grado 1 en la asociación.

Los detalles más importantes que se deben hacer notar respecto al Modelo Jerárquico, son los siguientes:

- En la gráfica que representa al modelo, no debe haber ciclos y sólo puede haber asociaciones 1:N y 1:1.
- Lo que en el modelo se representa gráficamente con flechas, son en verdad apuntadores o direcciones que deben mantenerse

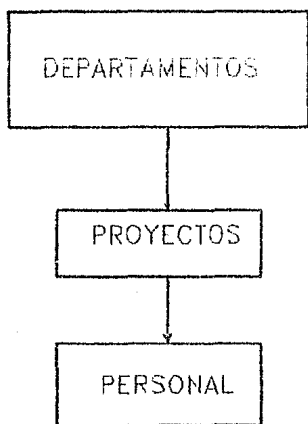


Figura 1.4 Diagrama de estructura de árbol.

junto con los datos.

- Al agregar, eliminar o recuperar información de la base de datos se debe "navegar" por las conexiones predeterminadas por las ligas, lo que implica un conocimiento detallado de la estructura de la base de datos.
- Si bien el "navegar" a través de apuntadores es muy rápido, se depende totalmente de la organización que se haya dado a la base de datos, para poder obtener información de ella. Cuando no se planea cierto tipo de consulta en el momento de diseño, lo más probable es que no se pueda obtener al explotarla.
- Para modificar la estructura de la base de datos, es necesario redefinir de nuevo todo el esquema.
- Se pueden representar las asociaciones M:N, pero esto hace necesario mantener datos duplicados.
- La forma de explotar la información es usualmente a través de programas de tercera generación.

Modelo de Red. El Modelo de Red (o reticular), al igual que el modelo Jerárquico, se representa como una gráfica. La diferencia radica en que para el modelo de Red se puede plantear una gráfica arbitraria (incluso ciclos) y no sólo conjuntos de árboles (Figura 1.5).

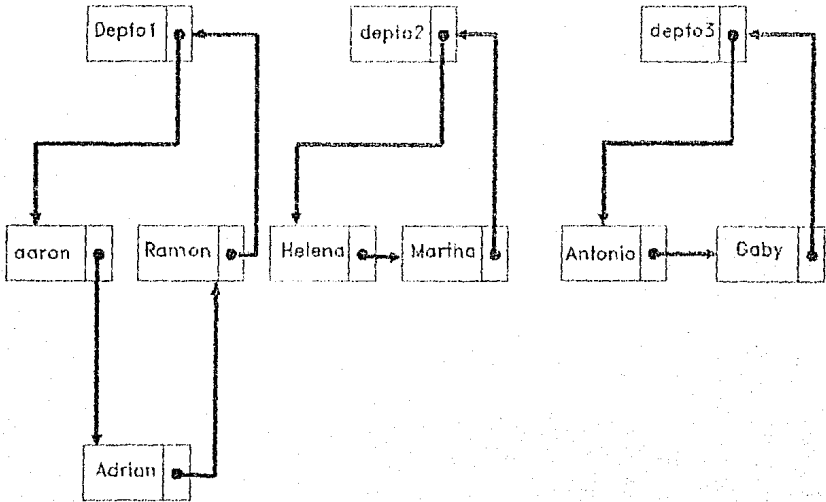


Figura 1.5 Modelo de red.

Tal como en el Modelo Jerárquico existe una ayuda gráfica, conocida como **diagrama de estructura de datos**, que se emplea al modelar. Los rectángulos representan tipos de registro, mientras que las líneas representan vínculos o ligas entre ellos (Figura 1.6).

Al difundirse este modelo y con el fin de crear un estándar, se formó el grupo DBTG² y definió algunas restricciones. Una de ellas es que solo se pueden utilizar asociaciones 1:N y 1:1, por lo que introdujo el concepto de conjuntos (DBTG Sets) para implantar el esquema.

Otra definición importante del estándar DBTG es el uso de grupos repetidos, lo cual permite que un campo pueda contener varios valores diferentes.

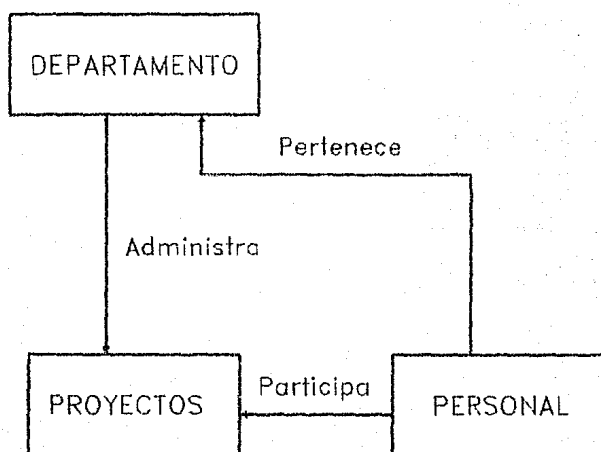


Figura 1.6 Diagrama de estructura de datos.

Por ser este modelo tan semejante al Jerárquico, las observaciones importantes son parecidas:

- En la gráfica que representa al modelo, no debe haber ciclos y sólo puede haber asociaciones 1:N y 1:1.

² El grupo DBTG (Data Base Task Group) de CODASYL definió en 1960 el primer estándar para Bases de Datos.

- Lo que en el modelo se representa gráficamente con flechas, son en verdad apuntadores o direcciones que deben mantenerse junto con los datos.
- Al agregar, eliminar o recuperar información de la base de datos se debe "navegar" por las conexiones predeterminadas por las ligas, lo que implica un conocimiento detallado de la estructura de la base de datos.
- Si bien el "navegar" a través de apuntadores es muy rápido, se depende totalmente de la organización que se haya dado a la base de datos, para poder obtener información de ella. Cuando no se planea cierto tipo de consulta en el momento de diseño, lo mas probable es que no se pueda obtener al explotarla.
- Para modificar la estructura de la base de datos, es necesario redefinir de nuevo todo el esquema.
- La forma de explotar la información es usualmente a través de programas de tercera generación.
- Se pueden representar relaciones M:N, pero esto va en contra del estándar y provoca redundancia.

Modelo Relacional. En 1969, tratando de simplificar el manejo de las bases de datos, E. F. Codd³ creó el modelo Relacional bajo la idea de que la forma de presentación de los datos hacia el usuario debía ser diferente a la forma en que éstos se almacenaban. Codd adoptó este modelo por la simplicidad de representación, a través de tablas que todos manejamos intuitivamente y por que a la vez se apoya en un fuerte fundamento matemático.

³ Edward F. Codd, de los Laboratorios de Investigación de IBM, publicó en 1970 un reporte en el que presentaba su idea. Después de eso se han producido un gran número de proyectos, iniciando con System R, que aplican su filosofía. El primer sistema relacional que apareció comercialmente fue ORACLE, y el lenguaje SQL se ha tomado como un estándar para la manipulación de bases de datos relacionales.

La primera diferencia respecto a los modelos anteriores, es que la información no se representa ni manipula a través de registros, sino que ahora se representa como un conjunto de relaciones (tablas) que contienen atributos (columnas), y que no contienen a otro tipo de información, como ligas o apuntes, que no sea la correspondiente a esas tablas (ver figura 1.7).

DEPARTAMENTOS

NUM.	NOMBRE
01	SOORTE TECNICO
02	AUTOMATIZACION
03	BASES DE DATOS
04	ESTADISTICA

PROYECTOS

CLAVE	NOMBRE
PROY 1	COMPILADOR DOP
PROY 2	ADAPIACION DE VOZ
PROY 3	USO DE SCANNER

PERSONAL

NUM.	NOMBRE	DEPTO.
01	NORBERTO	01
02	RAMON	02
03	GABRIELA	03
04	AARON	03

ASIGNACION

PERS.NUM.	PROY.CLAVE
01	PROY 2
03	PROY 3
04	PROY 2

Figura 1.7 Modelo relacional.

Los términos relación y atributo provienen de la base matemática del modelo, que es el álgebra relacional, mientras que los nombres de tablas y columnas se derivan de su representa-

ción gráfica. A continuación se muestra la analogía entre ambas notaciones:

<u>Algebra Relacional</u>	<u>Forma gráfica</u>
Relación	Tabla
Atributo	Columna
Tuplo	Renglón
Orden	Número de columnas
Cardinalidad, grado	Número de renglones

Del modelo relacional se pueden hacer algunas conclusiones:

- Con este modelo se pueden representar fácilmente las asociaciones M:N.
- No existen, o al menos el usuario así lo ve, apuntadores u otro tipo de información que no sea la que creó la necesidad de la base de datos.
- Las operaciones que se efectúan para obtener información de la base de datos, se realizan a nivel de la tabla completa y no a nivel de registros.
- Derivado del procesamiento a nivel de tablas, el tiempo de ejecución se ve incrementado en las operaciones relacionales.
- No es necesario diseñar el esquema de la base de datos de acuerdo a las operaciones o consultas que se van a realizar. Cuando los datos están en la base de datos es posible obtener información no prevista.
- Para modificar la estructura de la base de datos no es necesario alterar todas las definiciones de la misma. Incluso las aplicaciones no se ven afectadas por los cambios.

- La forma de explotar la información es a través de operaciones relacionales, normalmente implantadas en lenguajes de cuarta generación.

1.1.2.2 Modelos lógicos basados en objetos

Estos modelos son relativamente más recientes que los anteriores. Se caracterizan por ser muy flexibles e independientes de la forma en que los datos se almacenen y manipulan dentro de la computadora.

Los modelos basados en objetos continúan desarrollándose con vistas a hacer una representación más completa y versátil de la información, y se incluyen en los estudios relacionados con inteligencia artificial y bases de conocimiento. Sin embargo, con excepción de mencionar que existen modelos info-lógicos, semánticos y otros, solamente se hablará del Modelo de Entidades y Asociaciones⁴ por ser una técnica que, aunque se maneja con ciertas diferencias, se ha logrado definir más concretamente.

El Modelo de Entidades y Asociaciones se fundamenta en representar a la realidad como objetos independientes (Entidades) que mantienen vínculos (Asociaciones) con otros objetos. Los objetos se caracterizan por tener un nombre y un grupo de atributos que lo definen, por ejemplo:

ALUMNO(número de cuenta, nombre, carrera)

GRUPO(número, asignatura, cupo)

⁴ Al Modelo de Entidades y Asociaciones (MEA) se le conoce en inglés como "Entity-Relationship Model", sin embargo, al evitar traducir como Entidad-Relacionamiento se prefirió llamarlo así (o en ocasiones de Entidades y Vínculos), y por consiguiente se emplean las siglas DEA cuando se habla del Diagrama de Entidades y Asociaciones que en inglés es ERD.

Las Asociaciones indican un vínculo entre dos Entidades y algunas veces pueden tener además de un nombre uno o varios atributos:

ESTUDIA(número de cuenta, número)

De esta forma, podemos decir que un ALUMNO ESTUDIA en un GRUPO.

Para hacer un modelado de Entidades y Asociaciones, se cuenta con una herramienta gráfica conocida como Diagramas de Entidades y Asociaciones (DEA), la cuál utiliza rectángulos para representar conjuntos de Entidades, rombos para representar Asociaciones, líneas para indicar los vínculos entre Entidades, y en su forma completa, elipses para indicar los atributos. En los Apéndice (5A) se detallan formalmente los conceptos y mecanismos relacionados con esta forma de modelar.

Finalmente, se debe resaltar el hecho de la independencia que existe entre un Modelo de Entidades y Asociaciones, y la forma en que se almacenan los datos en la Base de Datos, lo cuál contrasta grandemente con los modelos basados en registros. Por otro lado, además de que el pasar de un Modelo de Entidades y Asociaciones a un Modelo Relacional es un proceso totalmente directo, es posible, con ciertos cambios, pasar también a un Modelo Jerárquico o Reticular.

1.1.3 Modelos y Esquemas

Cuando se diseña una Base de Datos, se trabaja con dos conceptos diferentes: el Modelo, que es una representación de la

realidad en una forma manejable, y el **Esquema**, que es la representación del Modelo bajo las características del sistema con que este último representará al modelo en la computadora. De esta forma podemos decir que un mismo modelo de Base de Datos tendrá diferentes esquemas al ser implantado bajo el sistema Oracle, que bajo el sistema Informix o que bajo Adabas, etcétera.

Por otro lado, la complejidad de un Sistema de Base de Datos nos lleva a tener que descomponerlo en elementos más simples. A esos elementos se les conoce como niveles de abstracción, y su finalidad es la de ocultar a los diferentes usuarios de una base de datos los aspectos ajenos a sus intereses, que pueden complicar su entendimiento o desempeño. Estos niveles se describen a continuación.

Nivel físico. es el de más bajo nivel de abstracción y en donde se definen cómo son realmente almacenados los datos, y cuáles son las estructuras de datos que los contienen.

Nivel conceptual. Es el nivel intermedio. Aquí está definido todo lo que contiene la base de datos en términos de estructuras relativamente simples y cuáles son los vínculos que existen entre los datos. En este nivel el Administrador decide qué debe estar dentro de la base de datos, aunque no requiere conocer los detalles de cómo están todos esos datos en el nivel físico.

Nivel de vistas. Es el nivel más externo. Aquí se describen sólo partes de la base de datos en base a todo lo que se encuentra en el nivel conceptual. Usualmente cada usuario tiene una vista diferente puesto que no requiere de todo lo que está

fuera de sus actividades, y así se mantiene ajeno a lo complejo que puede ser toda la variedad de información con que se cuenta y aún más lejano a la forma en que todo está almacenado en la computadora.

A cada uno de estos niveles de abstracción se le puede asociar un esquema equivalente, eso se muestra en la figura 1.8.

1.1.4 Manejadores de Bases de Datos

Es importante subrayar que, como se vio en las definiciones, la Base de Datos es un conjunto de información o datos relacionados y que se almacenan en algún medio. Aparte existe el Sistema Manejador de Base de Datos (SMBD), que es el software o conjunto de programas que se encargan de organizar y controlar el acceso a los datos. El SMBD es el responsable de proporcionar un ambiente conveniente y eficiente al tener acceso a la base de datos, para esto, entre otras cosas el SMBD debe desempeñar actividades como:

- interpretación de las consultas de los usuarios,
- interacción con el sistema administrador de archivos,
- vigilancia de la integridad de los datos,
- vigilancia de la seguridad de los datos,
- respaldo y recuperación,
- control de concurrencia.

El SMBD debe proporcionar además independencia de datos, esto es, la habilidad de hacer modificaciones en los diferen-

tes niveles de abstracción sin alterar significativamente los niveles complementarios. Los niveles de independencia son:

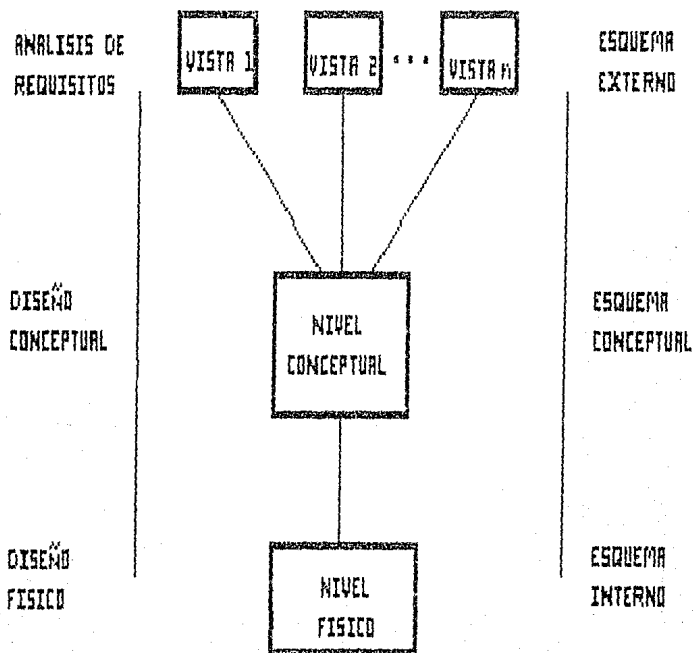


Figura 1.6 Niveles de abstracción.

- Independencia física de datos, que permite modificar al esquema físico sin obligar a que los programas de aplicación sean escritos de nuevo. Las modificaciones en el nivel físico se requieren principalmente por razones de mejorar el tiempo de respuesta del sistema.

¶ **Independencia lógica de datos**, que es la habilidad de modificar el esquema conceptual sin hacer que se tengan que modificar los sistemas de aplicación. Los cambios en el nivel conceptual se requieren cuando el modelo de la base de datos cambia, tal como cuando se agrega un atributo a una entidad.

Existen otros conceptos importantes relacionados con un Sistema Manejador de Base de Datos, que serán discutidos en esta sección.

1.1.4.1 Lenguajes

Tanto para definir los esquemas de una base de datos, como para explotar la información que contiene, se requiere de un lenguaje para la comunicación "hombre-sistema". Existen varios tipos de lenguajes de acuerdo a la actividad para la que son utilizados, y en algunos casos, todos están inmersos en uno solo.

LDD. Es el lenguaje de definición de datos (DDL por sus siglas en inglés). Se utiliza para crear los esquemas de la Base de Datos.

LMD. Es el lenguaje de Manipulación de datos (DML en inglés). Se aplica al recuperar, insertar o borrar datos de la Base de Datos. De éste además podemos decir que existen los llamados **procedurales**, con los que el usuario debe especificar el cómo los datos serán recuperados; y los **no procedurales**, que sólo requieren que el usuario especifique qué datos está solici-

En algunas instalaciones se cuenta con varios Administradores, uno por cada nivel de abstracción, que trabaja en lo individual sobre su área, pero deciden en grupo lo conveniente para la Base de Datos global.

1.1.4.3 Usuarios de la Base de Datos

Existen diferentes puntos de vista y diferentes aplicaciones que los usuarios hacen sobre la Base de Datos, es por eso que el SMBD debe contar con lenguajes de manipulación de datos que estén de acuerdo con el tipo de usuario involucrado. Los usuarios se pueden clasificar como:

- **Programadores de aplicaciones.** Profesionales de la computación que se dedican a la creación de programas de aplicación. Estos usuarios conocen y aplican posibilidades muy intrínsecas del Sistema Manejador.
- **Usuarios casuales.** Son usuarios que conocen un lenguaje de manipulación de datos e interactúan con el sistema sin escribir programas, haciendo consultas no predefinidas.
- **Usuarios finales.** Estos usuarios generalmente desconocen de computación, pero efectúan transacciones a través de programas de aplicación desarrollados especialmente para ellos.
- **Programadores especializados.** Son usuarios muy sofisticados que escriben aplicaciones no tradicionales, como bases de datos gráficas, para diseño, etcétera.

1.1.4.4 Estructura del Sistema

En conjunto, un Sistema de Base de Datos se compone de una variedad de módulos, algunos tienen que ver con los usuarios, otros con la organización de los datos, otros con los medios de almacenamiento, etcétera.

Si se trata de hacer una representación de todos los Sistemas de Bases de Datos, en general será como se muestra en la figura 1.9.

1.2 Modelado de la Base de Datos

Es sumamente importante contar con un buen Manejador de Base de Datos, dominar sus conceptos técnicos, y dominar también las técnicas de Bases de Datos. Sin embargo, atrás de todo eso hay algo todavía más importante: el modelado de la Base de Datos o Modelo de Datos que será manejado.

Cuando se diseñan bases de datos es necesario detenerse un momento y pensar en aquello que creó la necesidad fundamental de construir tales bases de datos y sus manejadores: la empresa, su información y todo cuanto la compone. La meta en el diseño lógico de Base de Datos es producir una Base de Datos lógica que satisfaga los requisitos de información de la Empresa.

La Base de Datos de la Empresa representa a la Empresa misma. Es por eso que antes de crear la Base de Datos es necesario hacer un modelo manejable de lo que contendrá. Al

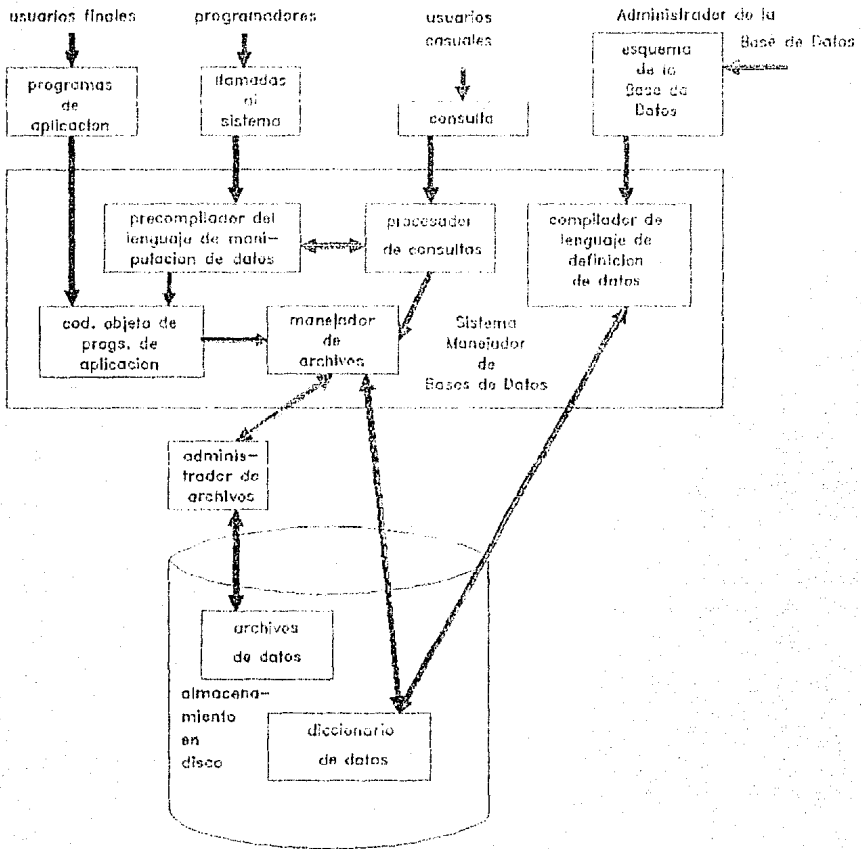


Figura 1.9 Representación de un Sistema de Bases de Datos.

desarrollar ese modelo, se debe tener en mente que lo que estamos haciendo es una pequeña representación de la misma empresa; se está convirtiendo a cada empleado, a cada movimiento, en un elemento de tal modelo.

Es entonces importante realizar el modelo de la base de datos desde una perspectiva global a la totalidad de la Empresa, aunque fijando los límites de la realidad.

1.2.1 Fronteras

Al iniciar el modelado se deben definir fronteras. Las fronteras crean límites que indican tanto "hasta donde" como "no menos de", con lo que se establecen marcas superiores de hasta dónde recabar información, para no distraerse con detalles inservibles; y marcas inferiores que lleven a buscar algo más.

1.2.1.1 Alcance de integración

En la etapa de definición de fronteras se genera la primera documentación, especificando qué estará dentro de la base de datos y por qué será incluido. Es la primera experiencia con los sistemas de la Empresa, aquí se comienza a entenderlos y al tratar de definirlos y justificarlos se descubre en ocasiones que realmente están fuera de las fronteras.

Al definir el alcance de integración se describe en idioma o lenguaje natural que datos y procesos estarán y cuáles no, dentro de la arquitectura conceptual.

1.2.1.2 Dimensiones

Es posible fijar diferentes niveles de fronteras, creando subconjuntos de la realidad total y simplificando la tarea por la vieja consigna de "divide y vencerás".

Una dimensión es una vista amplia de alguna parte de la Empresa que está siendo modelada; presenta la colección de diferentes partes que componen una misma actividad.

1.2.1.3 Vistas de datos

Al adentrarse en una dimensión, generalmente se ve que está formada por una variedad de aplicaciones. Cada aplicación o proceso se interesa únicamente en un pequeño grupo de los datos que componen la información de toda la Empresa. Ese grupo de datos que interesa a una aplicación se conoce como una vista de datos (Véase figura 1.10).

Una dimensión se compone de una o más vistas de usuarios. Una vista de usuario se compone, a su vez, de vistas de datos y vistas de procesos. Una vista de datos es cómo un usuario ve los datos dentro de una dimensión.

El alcance de integración se compone de diferentes dimensiones; diferentes vistas de usuarios hacen una dimensión; las vistas de datos son parte de las vistas de usuario.

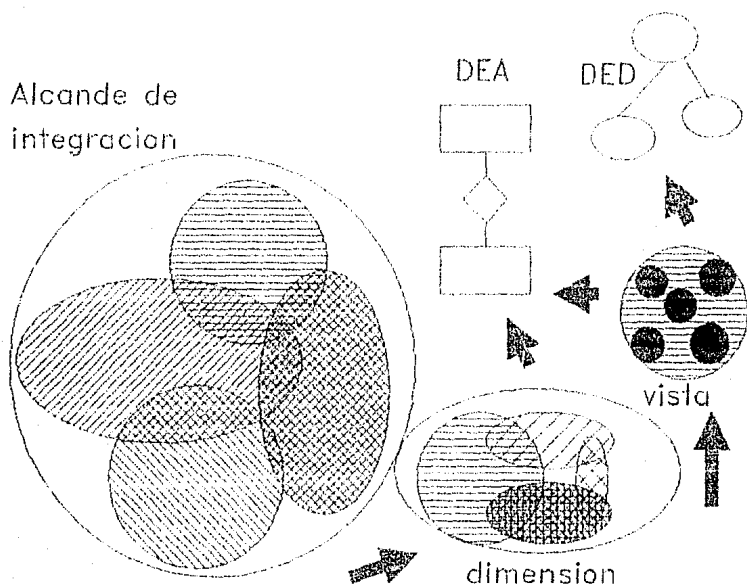


Figura 1.10 Alcance de Integración.

1.2.2 Búsqueda de requisitos

Habiendo determinado hasta donde llegará el modelo de la base de datos, y habiendo identificado a los elementos de la Empresa que la integrarán, en la siguiente etapa se debe obtener una lista con los elementos de datos que se van a manejar.

La determinación de requisitos requiere del estudio del sistema actual para conocer cómo trabaja y que mejoras deben ser hechas. Un requisito es una ventaja que debe ser incluida en un nuevo sistema. La determinación de requisitos significa el estudio de los sistemas existentes y la colección de detalles para encontrar éstos.

La etapa de búsqueda de requisitos implica recorrer todas las secciones de la empresa, que están dentro de nuestro alcance de integración, para conocer:

- los procesos que deben ser ejecutados,
- los requisitos de información de cada sección,
- los tiempos de ejecución requeridos,
- los volúmenes de información manejados,
- la forma en que se cumple actualmente con los sistemas,
- los controles que se llevan,
- qué datos son usados o producidos durante cada proceso,

etcétera.

1.2.2.1 Identificación de líderes

Antes de iniciar la búsqueda de requisitos, lo mejor es identificar primeramente a los líderes o jefes de usuarios. Los líderes que seleccionemos deben tener el control de al menos toda una dimensión, esto es, debe ser un miembro de una comunidad de usuarios responsable de coordinar las necesidades de los miembros de esa comunidad.

Se deben respetar los puntos de vista de cada líder y construir un modelo de datos de acuerdo a lo descrito por él; al líder se le debe dejar la responsabilidad de que el modelo creado represente al mundo real de su comunidad.

Para trabajar correctamente con cada líder, es sumamente importante que éste conozca la metodología de diseño que se seguirá, y por tanto, que la misma no sea compleja. Es nece-

carlo que el líder conozca el significado de los términos que son empleados.

Finalmente, se requiere que la comunidad de usuarios conozca la utilidad del modelo, puesto que éste representa a sus sistemas. No se deben hacer modificaciones al modelo sin la autorización del jefe de usuarios.

1.2.2.2 Entrevistas

Las entrevistas son utilizadas para coleccionar información verbalmente a través de cuestionarios propuestos por el analista. La habilidad del entrevistador es vital para el éxito de la entrevista. Se debe conocer el objetivo de ésta y las preguntas para el entrevistado específico.

De una entrevista esperamos dos tipos de información: cualitativa, como opiniones, políticas y descripciones narrativas de actividades o problemas; y cuantitativa, como números, frecuencias o cantidades.

Finalmente, teniendo en mente los objetivos se debe seleccionar qué tipo de entrevista debe emplearse, ya sea no estructurada --sesión de preguntas y respuestas libres-- para obtener información general, o estructurada --uso de preguntas normalizadas-- para obtener información específica.

1.2.2.3 Cuestionarios

El cuestionario provee una alternativa a la entrevista. Es la manera más fácil de contactar a muchas personas. Sin embargo, un buen cuestionario toma tiempo y arduo trabajo. El cuestionario se utiliza principalmente para coleccionar información cuantitativa.

Antes de utilizar el cuestionario se debe seleccionar a los receptores del mismo, y estar consciente de que las respuestas recibidas quizá no provengan de la persona seleccionada.

1.2.2.4 Observación

La observación es una técnica muy buena para percibir la realidad de las cosas. Es posible efectuar la observación de los procesos que componen a los sistemas actuales, y observar los registros que se utilizan en tales procesos.

La observación es muy útil cuando el analista necesita ver cómo son manipulados los documentos, cómo son ejecutados los procesos y si ocurren los pasos especificados. Sin embargo, el hecho de mirar a alguien puede cambiar la manera en que realiza su trabajo.

La observación de registros tiene menos riesgos, nos resume mucho estudio el revisar los libros y manuales de procedimientos, los documentos de comunicación y las formas que son capturadas o emitidas para y por los procesos.

1.2.3 Entidades y Asociaciones

El modelado de Entidades y Asociaciones es un proceso descendente usado para simplificar el procedimiento de diseño en circunstancias reales, en donde puede haber un gran número de atributos por considerar y más de una relación entre los atributos. La forma de crear un Modelo de Entidades y Asociaciones es a través de un Diagrama de Entidades y Asociaciones (DEA), el cual en forma gráfica muestra el comportamiento global de la Empresa.

Habiendo obtenido la información acerca de cuáles son los elementos de datos que intervienen en la Empresa que se está modelando, el siguiente paso es agregarlos para identificar Entidades (SA.2) y, con ayuda de la documentación que hasta el momento se ha logrado recabar, establecer las Asociaciones entre ellas. El paso final será colocar a cada elemento de datos --que realmente es un atributo de alguna Entidad-- en la Entidad que le corresponde.

Los pasos a seguir pueden describirse como a continuación se muestra:

1. Descubrir una Entidad.
2. Verificar que la entidad esté dentro del Alcance de Integración.
3. Determinar la llave candidato.
4. Documentar los resultados.
Repetir los pasos 1 al 4 según se requiera.
5. Descubrir Asociaciones.

6. Verificar que las Asociaciones están dentro del Alcance de Integración.
 7. Determinar los tipos de Asociación.
 8. Documentar los resultados.
- Repetir los pasos 5 al 8 según se requiera.
Regresar al paso 1, si es necesario reconsiderar.

Cuando la técnica se maneja más ágilmente, es una buena práctica el realizar los pasos anteriores en el mismo momento en que la información se obtiene del líder de usuarios, de esta forma las malas interpretaciones y las nuevas ideas pueden ser comentadas inmediatamente con los dueños del modelo. Además, en ese mismo instante se puede probar la utilidad del modelo para el usuario.

Usualmente los pasos anteriores se efectúan para cada una de las Dimensiones de la Empresa, obteniendo de este modo un conjunto de diagramas DEA que normalmente tienen elementos en común. El diagrama final se debe crear uniendo a todos esos diagramas independientes y eliminando las duplicidades y complementando las asociaciones entre Entidades.

1.2.4 Esqueletos y tablas normalizadas

El DEA se debe completar con los atributos que corresponden a cada Entidad o Asociación, para esto se crean listas identificadas con el correspondiente nombre de la Entidad o Asociación y entre paréntesis los atributos que la caracterizan.

Para crear ordenadamente tales listas, primeramente se construyen los Esqueletos. Estos esqueletos representan lo mínimo

que debe existir para garantizar que las asociaciones entre los datos se conserven. Tienen el nombre de la Entidad o Asociación, un atributo candidato a ser llave, y cuando se requiera, los atributos o llaves foráneas que mantendrán las asociaciones con otras Entidades.

Habiendo definido los esqueletos, es posible colocar a cada uno de los elementos de datos o atributos en la lista correspondiente, siguiendo los criterios de normalización (3B) establecidos para que la información de la base de datos se encuentre al menos en tercera forma normal.

La asignación de atributos se puede resumir como se muestra a continuación:

1. Descubrir un atributo.
2. Verificar que el atributo esté dentro del Alcance de Integración.
3. Encontrar el lugar del atributo, cumpliendo con las reglas de normalización.
4. Documentar los resultados.

Repetir los pasos 1 al 4 como sea necesario.

Regresar al paso 5 o al 1 de Entidades y Asociaciones, si es necesario reconsiderar.

Con frecuencia en la etapa de asignación de atributos es necesario regresar a la definición de Asociaciones o incluso de Entidades. Al hacer esto se debe pasar nuevamente por todas las etapas intermedias para revisar aquello que requiera de modificaciones por los cambios realizados. Si es necesario, se debe redibujar, reescribir y redocumentar todo lo que ha cambiado.

1.2.5 Definiciones

Uno de los aspectos más importantes en el modelado de una arquitectura conceptual es la definición de Entidades, Asociaciones y Atributos. Con una definición certera de datos al hacer el modelo se puede minimizar grandemente la ambigüedad y la inconsistencia.

Para calificar las definiciones, se pueden responder las siguientes preguntas:

- ¿La definición abarca todo lo que debiera abarcar?
- ¿La definición abarca sólo lo que debiera abarcar?
- ¿Existe alguna ocurrencia de datos en el Alcance de Integración, que no esté incluida en la definición?
- ¿Han sido definidos previamente todos los términos específicos del asunto?
- ¿Existe alguna ocurrencia de datos en el modelo, que ha sido definido para más de una Entidad?
- ¿Son comprensibles los términos usados en la definición?
- ¿El usuario y el diseñador de datos entienden y concuerdan con la definición?

Una forma práctica de documentar las definiciones de Entidades y Asociaciones es la de definir cada Entidad, y dibujar una tabla ejemplar que contenga ocurrencias típicas de sus atributos; bajo esta, definir las Asociaciones que se relacionen con la Entidad y en las que la primera parte de su llave primaria sea la llave primaria de la Entidad, dibujando la tabla ejemplar de cada Asociación.

Al definir los atributos es aconsejable hacerlo por orden alfabético, especificando el dominio que caracteriza a cada uno de ellos.

Con el fin de auxiliar al diseñador en la etapa de definición del Esquema Conceptual, el documento que culmine esta etapa de modelado puede incluir información acerca del mínimo, promedio y máximo número de ocurrencias que se estima mantener en un conjunto de Entidades, y la cantidad de elementos que contendrá cada conjunto de Asociaciones. Información como ésta es de gran ayuda en el momento de estimar el volumen de la Base de Datos.

Modelado

En este capítulo se describe el proceso de modelado, desde la definición del alcance de integración, hasta la presentación del modelo lógico completo que representa la base de datos.

En general se omiten los detalles de todo el trabajo, presentando únicamente el producto final de cada etapa del proceso.

2.1 Fronteras

Cerca del cien por ciento del volumen de información que maneja el Centro de Cálculo se relaciona con el control de alumnos, asignaturas, profesores e inscripciones, requerido por la Secretaría y la Oficina de Servicios Escolares, y por las Divisiones de la Facultad. Es por eso que el Alcance de Integración del modelo se define como la información que actualmente procesa el Centro con algún sistema de cómputo y que se mantiene en los medios magnéticos de la computadora principal del Centro.

Aunque el Centro de Cálculo no sea propiamente el usuario de la información, la responsabilidad de la misma se distribuye

entre las Coordinaciones que forman su organigrama, estando entonces dentro de esas Coordinaciones el verdadero control y conocimiento de la información. Es por eso que, para fines del modelado, las Coordinaciones del Centro se convierten en Dimensiones naturales dentro del Alcance de Integración.

Así pues, las Dimensiones tentativas fueron las siguientes:

- Coordinación del Anexo del CECAFI,
- Coordinación de Investigación y Desarrollo,
- Coordinación de Proyectos Académicos,
- Coordinación de Proyectos Administrativos,
- Coordinación de Planeación,
- Secretaría.

Finalmente, para llegar a las Vistas de Datos se hizo un inventario de los sistemas manejados por cada Coordinación, obteniéndose el siguiente catálogo, en donde se señala qué sistemas se encuentran dentro del Alcance de Integración:

Coordinación del Anexo del CECAFI

- Inventario de equipo de cómputo de la Facultad de Ingeniería (NO).
- Inventario de mobiliario y equipo del Centro de Cálculo (NO).

Coordinación de Investigación y desarrollo

- Actualización de historias académicas (SI).
- Generación de número de sorteo (SI).
- Dobles inscripciones (SI).

-
- ▣ Kardex (SI).
 - ▣ Resumen de profesores (SI).

Coordinación de Proyectos Académicos

- ▣ Nómina (SI).
- ▣ SICOVI (NO).
- ▣ Generación de recibos para becarios (NO).
- ▣ Asesorías del Centro (NO).
- ▣ Sistema de encuestas en cursos del Centro (NO).
- ▣ Automatización de Servicios Generales (NO).

Coordinación de Proyectos Administrativos

- ▣ Reinscripciones (SI).
- ▣ Extraordinarios (SI).
- ▣ Directorio de alumnos (SI).
- ▣ Servicios Escolares (SI).
- ▣ Encuestas (SI).
- ▣ Calificación de laboratorio del Departamento de Física (SI).
- ▣ Tesis (SI).
- ▣ Biblioteca (NO).
- ▣ SEFI (NO).
- ▣ Tutorías (SI).

Coordinación de Planeación

- ▣ Paquetería (NO).
- ▣ Estadística de paquetes (NO).
- ▣ Apoyo editorial (NO).
- ▣ MAEVA (NO).
- ▣ Inventario de Programas (NO).

- Estadísticas de consumo de recursos de cómputo (ND).

Secretaria

- Horarios (SI).
- Asistencia de profesores y académicos (SI).
- Servicio social (SI).
- Egresados (SI).
- Folios (SI).
- Direcciones de profesores (SI).

Cada Coordinación se hace cargo de un conjunto de sistemas y procesos (Vistas de Usuarios) que pertenecen a un mismo o diferente usuario. Las Vistas de Usuarios se componen de un conjunto de datos atómicos, mismos que forman las Vistas de Datos que nos dan la información necesaria para la creación del Modelo de Datos.

2.2 Búsqueda de requisitos

Una vez determinadas las Dimensiones y Vistas, la identificación de líderes se dio por sí sola, puesto que los Coordinadores tienen de alguna forma el control completo de toda su Dimensión.

Para cada Coordinador se planearon una serie de entrevistas. Las primeras fueron no estructuradas, para darle la información acerca de qué era lo que se estaba haciendo y cuál era el método a seguir; aquí se definieron los sistemas que pertenecen al Alcance de Integración propuesto --los cuales se indican en la lista anterior con (SI)-- y se descartaron los que se

consideraron fuera de ese alcance --se indican con (NO). Posteriormente con entrevistas estructuradas se obtuvo información específica acerca de cada proceso, periodos de ejecución, archivos utilizados, volúmenes, permanencia de información, atributos, definiciones de éstos, sinónimos, ocurrencias, tipo de valor, representación, valores permitidos e información adicional de utilidad para catalogar y cuantificar los datos atómicos de las Vistas. Finalmente se procedió a efectuar nuevamente entrevistas no estructuradas, con el fin de crear Diagramas de Entidades y Asociaciones que representarían las actividades de cada Dimensión.

Para complementar la compilación de información necesaria para llegar al modelo de Entidades y Asociaciones, se analizaron documentos como manuales de procedimientos, descripciones de registros, solicitudes y formas de captura, tablas de códigos, etcétera.

2.3 Entidades y Asociaciones

El análisis de datos condujo a obtener un conjunto de Diagramas de Entidades y Asociaciones que servían a los propósitos de cada Dimensión. Todos esos diagramas fueron unidos en uno solo, eliminando duplicidades y modificando en algunos casos las Asociaciones para que se adaptaran en la mejor forma al modelo resultante. El diagrama se ilustra en la figura 3.1.

Fueron eliminadas algunas Entidades y Asociaciones después de concluir que se habían generado por la forma en que los procesos que a esa fecha se utilizaban manejaban la información. "ACTUALIZACION", "PROCEDENCIA" y "DIA" son en realidad

Modelado

Entidades Secundarias, puesto que no pertenecen a la realidad del Alcance de Integración que se definió para esta Base de Datos. Sin embargo, se definieron por la conveniencia de reducir la repetición de información y para hacer más claro al modelo. Además, Asociaciones como "Inscrito" y "AltaBajaCambio" se incluyeron por la conveniencia de tenerlas dentro del modelo, a pesar de que representan información temporal de entrada a procesos que la transforman y dejan en su estado final en algún otro lugar.

La unión de los conjuntos de entidades "ALUMNO INTERNO" y "ALUMNO EXTERNO" produce el conjunto de alumnos que se inscriben a asignaturas ("ALUMNO"), lo que define la generalización mostrada en el diagrama. Por otro lado, de "ALUMNO INTERNO" se tomó un subconjunto de entidades formado por los alumnos de nuevo ingreso y se creó la entidad "ALUMNO NUEVO INGRESO" a través de una especialización.

Por último, la Entidad "ACTUALIZACION", que no tiene vínculo con ninguna otra Entidad del modelo, se incluyó para mantener información global y permanente que, de otra forma, se repetiría en grandes volúmenes en algunas Entidades de la Base de Datos.

Fue necesario modificar algunas veces el modelo, pues mientras los líderes de usuarios comprendían mejor su función, se producían más cambios y mejoras, llegando finalmente al diagrama ya señalado en la figura 3.1.

2.4 Esqueletos y tablas normalizadas

Siguiendo las reglas para la obtención de esqueletos, a partir del diagrama general de Entidades y Asociaciones se obtuvo la siguiente lista:

2.4.1 Esqueletos

ACADENICO(id-académico,...,
 ACTUALIZACION(id-fecha-de-actualización,...,
 ALUMNO(id-alumno,...,
 ALUMNO EXTERNO(id-alumno,...,
 ALUMNO INTERNO(id-alumno,...,
 ALUMNO NUEVO INGRESO(id-alumno,...,
 ASIGNATURA(id-asignatura,...,id-departamento,...,
 CARRERA(id-carrera,...,id-órgano,...,
 DEPARTAMENTO(id-departamento,...,id-órgano,...,
 DIA(id-día,...,
 EDIFICIO(id-edificio,...,
 GRUPO(id-grupo,...,id-asignatura,...,
 GRUPO EXTRA
 (id-grupo-extraordinario,...,id-asignatura,...,id-salón,...,
 GRUPO TUTOR(id-grupo-tutor,...,id-académico,...,
 LUGAR DE ADEUDO(id-lugar-adeudo,...,id-departamento,...,
 NOMINA(id-nomina,...,
 ORGANO(id-órgano,...,
 PROCEDENCIA(id-procedencia,...,
 SALON(id-salón,...,id-órgano,...,id-edificio,...,
 TESIS(id-tesis,...,

Adeuda(id-alumno, id-lugar-adeudo, ...,
Alta baja cambio(id-alumno, id-grupo, ...,
Asignación()
Asignación extraordinaria()
Cita(id-grupo-tutor, id-día, ...,
Clase(id-grupo, id-académico, id-día, id-salón, ...,
Compone(id-asignatura, id-carrera, ...,
Contrata(id-departamento, id-académico, ...,
Coordina()
Dueño()
Equivalencia(id-asignatura_{antigua}, id-asignatura_{actual}, ...,
Estudia(id-alumno, id-carrera, ...,
Examen(id-alumno, id-tesis, ...,
Examina()
Falta(id-académico, id-grupo, ...,
Ha estudiado(id-alumno, id-carrera, ...,
Inscrito(id-alumno, id-grupo, ...,
Inscrito extra(id-alumno, id-grupo-extraordinario, ...,
Ofrece()
Org-Dep()
Paga(id-nómina, id-académico, ...,
Pertenece()
Presentó(id-alumno, id-asignatura, ...,
Proviene()
Seriación(id-asignatura_{antecedente}, id-asignatura_{consecuente}, ...,
Sinodal(id-grupo-extraordinario, id-académico, ...,
Solicita(id-alumno, id-grupo, ...,
Tutor()
Tutoría(id-alumno, ..., id-grupo-tutor, ...,
Ubica()

2.4.2 Tablas y atributos

Es conveniente mencionar algunas consideraciones que se tomaron para la definición de llaves primarias y asignación de atributos.

Como antes se mencionó, la entidad "ACTUALIZACION" no es parte de la realidad modelada, fue creada con la finalidad de tener dentro de la base de datos las fechas de actualización de la información y sólo contiene un tuplo y dos atributos poco relacionados. Así pues, no requiere de una llave primaria como identificador.

Existen algunas ideas que surgieron como obvias: una buena llave primaria para "DEPARTAMENTO" es la concatenación de la clave del Órgano con la clave del departamento correspondiente; esto se debe a que los números de departamento se repiten para cada Órgano, así pues:

$$\text{id-departamento} = \text{id-órgano} + \text{número-de-departamento}$$

Por otro lado, una buena llave primaria para "GRUPO" sería la concatenación de la clave de la asignatura que se imparte en él, con el número de grupo correspondiente, lo que se debe a la misma razón descrita para la entidad "DEPARTAMENTO". De este modo:

$$\text{id-grupo} = \text{id asignatura} + \text{número-de-grupo}$$

Para identificar a un grupo de tutorías ("GRUPO TUTOR"), se puede crear una clave utilizando el atributo "grupo-tutor",

asignado por la DGAE a cada alumno de nuevo ingreso. El atributo "grupo-tutor" es un número que se repite cada periodo de nuevo ingreso, por lo que para crear una llave primaria se le debe agregar el año de ingreso del alumno a la Facultad. De tal forma quedaría como:

id-grupo-tutor = Año-ingreso + Grupo-tutor

Para identificar a un grupo extraordinario, se podría utilizar lo siguiente:

id-grupo-extraordinario = id-asignatura +
número-de-grupo-extra

En el caso de "LUGAR DE ADEUDO":

id-lugar-adeudo = id-departamento +
clave-lugar-de-adeudo

Entre las entidades "ALUMNO INTERNO" y "ASIGNATURA" pueden existir varias ocurrencias de tipo "Presentó" que relacionen a iguales ocurrencias de esas entidades, es por eso que se debe agregar un atributo a la llave primaria con el fin de que sea única. El problema se resuelve concatenando el folio del acta (folio del grupo) en que se asienta las calificaciones de ese grupo. El identificador id-nómina sería también una llave compuesta, puesto que un académico puede tener varios registros en el archivo. Se define como:

id-nómina = registro federal de causante +
número en nómina

Sin embargo, la creación de este tipo de llaves primarias, aunque bastante razonables y de acuerdo con las tradiciones y costumbres del Centro de Cálculo, no es muy correcta. La fabricación de restricciones ficticias y dependencias indeseables de actualización podrían acarrear problemas posteriores.

Por motivos como los expuestos anteriormente, se crearon llaves primarias definidas por el sistema, conservando como atributos, en donde así conviene, los elementos de las llaves candidato compuestas, definiendo de esta manera únicamente tablas primas para las entidades. De tal forma, las tablas surgidas de la primera etapa de diseño son las mostradas a continuación (Los atributos que forman la llave primaria de cada tabla, se presentan subrayados).

ACADEMICO

registro federal de causante
título
nombre del académico
folio
estado de contratación

ACTUALIZACION

fecha de actualización alumnos
fecha de la nómina

ALUMNO

número de cuenta
número de sorteo
créditos cursando
número de teorías inscrito
número de laboratorios inscrito
tipo de alumno

ALUMNO EXTERNO

número de cuenta
nombre del alumno
clave escuela de procedencia

ALUMNO INTERNO

número de cuenta
nombre del alumno
domicilio
colonia/población
delegación/municipio
código postal
clave del estado
teléfono
fecha de nacimiento
clave de procedencia (nacionalidad)
sexo
año de ingreso a DGAC
año-semestre de ingreso a la facultad
créditos cubiertos
créditos revalidados
inscripciones en ordinario
asignaturas aprobadas en ordinario
asignaturas aprobadas en extraordinario
asignaturas reprobadas

promedio
índice de escolaridad
índice de velocidad
irregularidad
posible consejero
semestres sin cursar asignatura
último año-semestre en cursar asignatura
año-semestre presentó último extraordinario
número de movimientos en H.A.

ALUMNO NUEVO INGRESO

número de cuenta
nombre del alumno
domicilio
colonia/población
delegación/municipio
código postal
clave del estado
teléfono
fecha de nacimiento
clave de procedencia (nacionalidad)
sexo
grupo primer ingreso
plan escolar asignado

ASIGNATURA

clave de asignatura
nombre de la asignatura
créditos
laboratorio o práctica
año-semestre de inicio
vigencia
clave de departamento

CARRERA

clave de la carrera
nombre de la carrera
clave del órgano
créditos actuales

DEPARTAMENTO

clave de departamento
clave del órgano
número de departamento
nombre del departamento

DIA

número de día
nombre del día

EDIFICIO

clave de edificio
ubicación de edificio

GRUPO

número único de grupo
número de grupo
semestre grupo
cupo
vacantes
clave de asignatura

GRUPO EXTRA

número único de grupo extraordinario
número de grupo extraordinario
período
fecha y hora de extraordinario
número de salón
clave de asignatura

GRUPO TUTOR

número de grupo tutor
año de inicio tutoría
grupo tutor
primer salón
lugar de tutoría
registro federal de causante

LUGAR DE ADEUDO

clave de lugar de adeudo
nombre de lugar de adeudo
clave de departamento

NUMINA

número en nómina
registro federal de causante
nombre en nómina
nombre de categoría
clave de categoría
nivel
número de plaza
sueldo
programa-subprograma
dependencia-subdependencia
partida
digito de control
años de servicio académico
fecha de inicio de servicio académico
quincenas pagadas
mes alta administrativos
año alta administrativos
fecha de alta académicos
fecha límite de contrato académico
clave desconocida
horas académicos

ORGANO

clave del órgano
nombre de órgano
sigla

PROCEDENCIA

clave de procedencia
nombre de procedencia

SALON

número de salón
clave del órgano
proyección
capacidad
clave de edificio
piso a nivel

TESIS

clave de tesis
nombre de la tesis

Modelado

Adeuda

número de cuenta
clave de lugar de adeudo
fecha de adeudo

Alta baja cambio

número de cuenta
número único de grupo alta
número único de grupo baja
movimiento
opción

Cita

número de grupo tutor
número de día
hora inicial de asesoría
hora final de asesoría

Clase

numero unico de grupo
registro federal de causante
número de día
número de salón
tipo de clase
número de profesor
hora inicial
hora final

Compone

clave de la carrera
clave de asignatura
semestre
obligatoriedad

Contrata

clave de departamento
registro federal de causante

Presentó

número de cuenta
clave de asignatura
folio del grupo
año-semester de evaluación en H.A.
tipo de examen
calificación

Seriación

clave de asignatura anterior
clave de asignatura consecuente

Sinodal

número único de grupo extraordinario
registro federal de causante
primero o segundo

Solicita

número de cuenta
número único de grupo
opción

Tutoría

número de cuenta
número de grupo tutor

2.5 Definiciones

La etapa final de modelado fue la organización de las definiciones de cada Entidad, Asociación y Atributo, terminando de esta forma la documentación necesaria para crear el modelo conceptual requerido para el manejador de base de datos seleccionado.

En esta sección se muestran las definiciones de las Entidades y Asociaciones presentes en el Modelo, las marcadas con un asterisco son Asociaciones de las que no se obtiene una tabla. Posteriormente vienen las definiciones de atributos, incluyendo el dominio de donde cada uno de éstos toma su valor o valores. Más adelante se muestran las tablas ejemplares, mismas que permitieron formar una idea más precisa de lo que se estaba modelando.

Se muestran dos tablas cruzadas en donde se indica la asociación y el orden de precedencia entre Entidades y Asociaciones. Estas tablas son de mucha utilidad, pues definen claramente la dependencia que existe entre las diferentes Entidades. Los resúmenes de los nombres de las Entidades, Asociaciones y Atributos que participan en el Modelo se presentan en otro capítulo (95.2), puesto que fueron realizados como un auxilio en la creación del esquema conceptual.

Para finalizar se creó una tabla en la que se describe el tipo de acceso que hace cada Vista --en este caso, cada Coordinación-- de cada uno de los atributos que forman el modelo de datos. Esa tabla será de utilidad para la posterior administración de la base de datos, pues ayudará en la creación

del esquema externo y en la delimitación de responsabilidades y derechos.

2.5.1 Definición de Entidades

ACADEMICO

Personal de la Facultad, contratado en alguna plaza para actividades académicas por algún departamento de las Divisiones. Pueden existir académicos actualmente no contratados.

ACTUALIZACION

Puesto que la información de la entidad "ACIGIATURA" y la Asociación "Presentó" se actualiza sólo hasta que la Dirección General de Administración Escolar (DGAE) envía a CECAFI la cinta de archivo maestro, en esta entidad se mantiene la fecha (año-semester) hasta la que es válida esa información. Por otro lado, la fecha de la información de nomina que se tiene aparece también en esta entidad.

ALUMNO

Estudiante de la UNAM, que cursa asignaturas en la Facultad de Ingeniería. A un alumno en esta entidad ya se le ha generado un número de sorteo para su reinscripción semestral.

ALUMNO EXTERNO

Estudiante de la UNAM, que no está registrado en el plantel 011, y que se inscribe para cursar asignatura de la Facultad de Ingeniería. Su historia académica no se lleva en esta Facultad.

ALUMNO INTERNO

Estudiante de la UNAM, registrado en el plantel 011, que: o bien estuvo inscrito en la Facultad de Ingeniería pero no se ha titulado, o está actualmente inscrito en una o varias asignaturas, o estuvo inscrito y ya se tituló. Reconocido por la DGAE.

ALUMNO NUEVO INGRESO

Estudiante que cursa asignaturas en la Facultad de Ingeniería y del cual aun no se recibe información académica histórica de DGAE por ser de reciente ingreso.

ASIGNATURA

Asignatura que está o estuvo contemplada en el plan de estudios de alguna carrera de la Facultad. Asignaturas no vigentes pueden tener equivalentes con las vigentes actuales y viceversa. Existen asignaturas que no pertenecen al plan de estudios de alguna carrera, pero que la administración escolar las utiliza.

CARRERA

Area profesional de Licenciatura, que se ofrece en esta Facultad. Se coordina por una División.

DEPARTAMENTO

Elemento que pertenece al organigrama de un Organó, en el nivel bajo la Jefatura. Se encarga de coordinar asignaturas y académicos a su cargo.

DÍA

Día de la semana: lunes, martes, miércoles, jueves, viernes, sábado, domingo.

EDIFICIO

Lugar dentro de las instalaciones de la Facultad de Ingeniería en donde se encuentran grupos de salones y laboratorios.

GRUPO

Opción de una asignatura, con uno o mas profesores ("ACADÉMICO") a su cargo, dentro de un horario, en el cual los alumnos se inscriben para cursar una asignatura.

GRUPO EXTRA

Grupo que es abierto para inscripción a examen extraordinarios. Los exámenes en estos grupos se realizan en alguno de los cuatro periodos de fechas en un semestre.

GRUPO TUTOR

Organización que agrupa a un conjunto de alumnos que tienen a un mismo profesor de la Facultad de Ingeniería como tutor.

LUGAR DE ADEUDO

Departamento, laboratorio, biblioteca, o lugar en que se da un servicio de préstamo de material, y en el que un alumno se puede encontrar como deudor de material.

NOMINA

Es el conjunto de registros de los trabajadores académicos y administrativos que pertenecen a la Facultad de Ingeniería. La información se recibe periódicamente de la Dirección General de Servicios de Cómputo Administrativo.

ORGANO

Elemento de la organización de alto nivel de la Facultad de Ingeniería. Algunos órganos son las Divisiones, responsables de organizar y administrar la estructura, planes de estudio, asignaturas y personal académico de las carreras profesionales de licenciatura. También se incluyen Secretarías, Centros de servicio y otros.

PROCEDENCIA

Nombre del Estado de procedencia, para alumnos nacionales, o nacionalidad, para alumnos extranjeros, de la Facultad.

SALON

Aula o laboratorio en que se imparte una asignatura de la Facultad de Ingeniería. Cada salón cuenta con un grupo de características, como el ser apto para proyección, lugar de ubicación, etcétera.

TESIS

Trabajo escrito que un alumno presenta como requisito para la obtención de su título profesional. Por reglamento, este trabajo escrito se debe defender en un EXAMEN profesional oral.

2.5.2 Definición de Asociaciones

Adsuda

Indica los lugares de adsudo en que un alumno puede tenerlo.

Alta baja cambio

Grupo que un alumno solicita en segundo intento de inscripción o bien, para baja definitiva o condicionada a la alta en otro grupo.

Asignación *

Vincula a un grupo con la asignatura que se imparte en tal grupo.

Asignación extraordinaria *

Vincula a un grupo extraordinario con la asignatura que será presentada en ese grupo.

Cita

Vincula a cada grupo tutor con los días de la semana en que el tutor atiende a sus alumnos tutoriados.

Clase

Para cada grupo en que se ofrece una asignatura, esta asociación indica por cada día de la semana en que tiene clase, el académico encargado de su impartición y el horario correspondiente.

Compone

Determina el nombre y características de las asignaturas que componen la retícula de cada carrera. Aquí se definen las características de la asignatura para cada carrera.

Contrata

Define el o los departamentos de los Organos en que un académico es contratado.

Coordina *

Relaciona al departamento de una División, con las asignaturas que coordina.

Dueño *

Relaciona los salones y laboratorios de la Facultad de Ingeniería, con el Organó al que corresponde.

Equivalencia

Puesto que los planes de estudio han venido cambiando, y las asignaturas reestructurándose y variando sus nombres, existe una equivalencia entre algunas asignaturas anteriores y algunas asignaturas actuales.

Estudia

Relaciona a un alumno con la carrera (primera, segunda o simultanea) en que se encuentra inscrito dentro de la Facultad de Ingenieria. De acuerdo al reglamento, una carrera simultanea es la carrera a la que se inscribe un alumno estando aún cursando una primera carrera; o después de haber terminado una primera carrera sin haberse titulado. Una segunda carrera es a la que se inscribe un alumno, habiéndose titulado en una anterior. En ambos casos, el alumno tiene el 150% del tiempo normal para terminar esa carrera, contado a partir de su inscripción en ella.

Examen

Acto en que uno o varios estudiantes se presenta a defender una tesis profesional, para obtener título de alguna carrera.

Examina

Asociación que vincula a un grupo extraordinario con el salón en el que se efectuará.

Falta

Mantiene la información de las inasistencias de un profesor o académico a su jornada de trabajo o clase.

Ha estudiado

Relaciona a un alumno de la Facultad con otra u otras carreras en las que también se ha inscrito, pero ya no pertenece a ellas por haberse titulado, o haber cambiado de carrera. Un cambio de carrera consiste en renunciar a una carrera para inscribirse en otra. El alumno cuenta con un 150% del tiempo normal de terminación para acabarla, contado a partir de la inscripción a su primera carrera.

Inscrito

Indica en que grupos queda inscrito un alumno, para un semestre.

Inscrito extra

Indica en que grupos extraordinario se inscribe un alumno, para presentar un examen.

Ofrece *

Relaciona cada carrera que existe en la Facultad de Ingeniería, con la División que se encarga de coordinarla.

Org-Dep *

(Órgano-Departamento) Vincula a cada Órgano con sus departamentos.

Paga

Relaciona los registros del archivo de nómina de la facultad de Ingeniería, con los miembros del personal académico a los que se les paga por nómina.

Pertenece *

Nos muestra el Departamento al que pertenece un lugar de aducido.

Presentó

Vincula a un alumno con las asignaturas en que se ha inscrito. Contiene la información de la calificación obtenida, el tipo de examen presentado, y otra.

Proviene *

Vincula a un alumno de la Facultad con el Estado del que proviene, o con su nacionalidad en caso de alumno extranjero.

Seriación

Establece, para una asignatura, cuales se sugieren cursar antes y después de ella, según el último plan de estudios aprobado para la carrera.

Sinodal

Para cada grupo extraordinario, se tienen profesores propuestos como primero y segundo sinodal.

Solicita

Grupo que solicita un alumno en primer intento de inscripción, para cursar una asignatura. Existen dos opciones de grupo para cada asignatura solicitada, que pueden ser iguales.

Tutor *

Vincula a un académico con el grupo o los grupos de tutoría que tiene asignados.

Tutoría

Vincula a un alumno con el grupo de tutoría asignado a él por el Centro de Servicios Educativos (CESEFI).

Ubica *

Vincula a cada salón con su correspondiente edificio.

2.5.3 Definición de Atributos

año de alta administrativo

Representa el año en que un empleado universitario fue dado de alta como trabajador, con plaza administrativa. Campo numérico de 2 dígitos.

año de ingreso a DGAE

Año en que el alumno está registrado ante DGAE (sólo a nivel licenciatura o superior). Únicamente se tienen los dos últimos dígitos del año, los primeros dos son siempre 19 y se toman intrínsecamente.

año de inicio tutoría

Año en que un alumno ingresa al sistema de tutorías y se le asigna un tutor y un grupo de tutoría. Campo numérico de 2 dígitos.

año-semestre de evaluación en H.A.

Año y semestre en el que se realizó el movimiento en Historia Académica, que se describe para el alumno. Cadena de 3 caracteres numéricos

AAS	en donde:
AA	año
S	semestre (1 ó 2)

año-semestre de ingreso a la Facultad

(año de ingreso) Año y semestre en que tuvo el alumno su primer movimiento como alumno de la Facultad de Ingeniería. Campo numérico de 3 dígitos.

año-semestre de inicio

Periodo o año-semestre en que se comenzó a impartir una asignatura. Campo numérico de 3 dígitos.

AAS	en donde:
AA	año
S	semestre (1 ó 2)

año-semester de registro a carrera

Año y semestre en que se registró al alumno como inscrito en la carrera.

Cadena de tres caracteres numéricos.

AAS en donde:
AA dos últimos dígitos del año
S semestre (1 o 2)

año-semester presentó último extraordinario

Año (dos últimos dígitos) y semestre en que el alumno realizó su última inscripción a examen extraordinario.

Campo numérico de 3 dígitos.

años de servicio académico

Sólo para el personal académico, representa su antigüedad académica.

Campo numérico de 2 dígitos.

asignaturas aprobadas en extraordinario

Total de asignaturas que el alumno ha aprobado con examen extraordinario.

Campo numérico entero de 3 dígitos.

asignaturas aprobadas en ordinario

Total de asignaturas a las que el alumno se ha inscrito, habiendo aprobado examen en el mismo periodo en que se inscribió.

Campo numérico entero de 3 dígitos.

asignaturas reprobadas

Total de asignaturas que el alumno ha presentado en examen y no ha acreditado, ya sea en periodo ordinario o extraordinario.

Campo numérico entero de 3 dígitos.

calificación

Código que representa la calificación obtenida por un alumno al presentar un examen ordinario o extraordinario, de una asignatura.

Carácter alfanumérico, con los siguientes posibles valores:

1	HE
2	B
3	S
4	NA
A	10
B	acreditado
C	revalidado
D	no presente
9	9
8	8
7	7
6	6

capacidad

Cupo estimado para un salón de clase o laboratorio, principalmente definido por el número de butacas con que ése cuenta.

Valor numérico entero de tres dígitos.

clave de asignatura

Número asignado a cada asignatura que es o fue impartida en las carreras de la Facultad de Ingeniería.

Cadena numérica de 4 caracteres.

Mmmm en donde:

M	0	Asignatura teórica.
	1	Laboratorio.

mmm Número correspondiente a la asignatura.

clave de asignatura_{actual}

Número que identifica a una asignatura de los planes de estudios actuales, que es equivalente a una asignatura descrita por el atributo clave de asignatura_{antigua}.

Tiene las mismas características que el atributo clave de la asignatura.

clave de asignatura_{antigua}

Número que identifica a una asignatura que es equivalente a otra en los planes de estudios actuales.

Tiene las mismas características que el atributo clave de asignatura.

clave de asignatura ^{anterior}
Clave de una asignatura que según el plan de estudios propuesto, se debe cursar antes que otras (se refiere a la seriación indicativa).
Tiene las mismas características que el atributo clave de asignatura.

clave de asignatura ^{consecuente}
Clave de una asignatura que según el plan de estudios propuesto, se debe cursar después que otra(s).
Tiene las mismas características que el atributo clave de asignatura.

clave de categoría
Número que se asigna a cada plaza de Trabajo, según los reglamentos y normas de la Facultad de Ingeniería.
Cadena alfanumérica de 4 caracteres.

clave de departamento
Número serial que identifica en forma única a cada Departamento.
Campo numérico de 2 dígitos con inicio en 01.

clave de edificio
Número serial que identifica en forma única a cada edificio de la Facultad de Ingeniería.
Campo numérico de 2 dígitos.

clave de la carrera
Código de la carrera que se imparte o fue impartida en la Facultad de Ingeniería.
Campo numérico de 2 dígitos, con los siguientes valores:

21	Ingeniero Civil.
22	Ingeniero de Minas y Metalurgista.
23	Ingeniero Geólogo.
24	Ingeniero Petrolero.
25	Ingeniero Topógrafo y Geodesta.
26	Ingeniero Mecánico Electricista.
27	Ingeniero Mecánico.
28	Ingeniero Industrial.
29	Ingeniero Eléctrico y Electrónico.
31	Ingeniero Geofísico.
32	Ingeniero en Computación.

y otros de carreras no vigentes.

Modelado

clave de lugar de adeudo

Código serial que identifica a cada uno de los laboratorios o bibliotecas en que se ofrece un servicio de préstamo, y un alumno puede ser deudor.

Campo numérico de dos dígitos que inicia en 01.

clave de procedencia

(nacionalidad) Código que identifica a cada uno de los Estados de la República, de donde puede provenir un alumno, y a cada una de las nacionalidades que puede tener un alumno extranjero.

Cadena numérica de 3 dígitos.

clave de tesis

Número serial asignado por el CECAFI a cada uno de las tesis profesionales que se presentan en la Facultad.

Campo numérico entero de 6 dígitos.

clave del estado

Número asociado con el nombre de alguno de los estados que circundan al D.F., o el mismo D.F.

Cadena de 2 caracteres numéricos, con el siguiente significado:

01	Distrito Federal
02	Estado de México
03	Morelos
04	Puebla
05	Hidalgo
06	Tlaxcala

clave del órgano

Código que representa a uno de los miembros de alto nivel en la organización de la Facultad, como Secretarías, Divisiones y Centros. Campo numérico de 2 dígitos, con valores:

0	Dirección,
1	Secretaría General,
2	División de Ingeniería Civil, Topográfica y Geodésica,
3	División de Ingeniería Mecánica y Eléctrica,
4	División de Ingeniería en Ciencias de la Tierra,
5	División de Ciencias Básicas,
6	División de Ciencias Sociales y Humanidades,
7	División de Estudios de Posgrado,
8	División de Educación Continua,
9	Centro de Cálculo,
10	Centro de Servicios Educativos,
11	Secretaría de Servicios Escolares,
12	Coordinación de Servicios Generales,
13	Coordinación de Bibliotecas,
14	Secretaría Administrativa,
15	Secretaría de Asuntos Estudiantiles,
16	Unidad de Planeación
17	Unidad de Apoyo Editorial.

clave desconocida

Clave desconocida, utilizada en la nómina.
Cadena numérica de 3 caracteres.

clave escuela de procedencia

Código asignado por la UNAM a sus planteles. Se utiliza en el manejo de estudiantes que provienen de escuelas que no son OII Facultad de Ingeniería.
Cadena numérica de 3 caracteres.

código postal

Número que identifica la zona de servicio postal de la residencia del alumno (CP).
Cadena de 5 caracteres numéricos.

colonia/población

Para alumnos que radican en el D.F., colonia de residencia.
Para alumnos que no radican en el D.F. es el nombre de la población o ciudad.
Cadena de 30 caracteres alfanuméricos.

créditos

Valor académico de una asignatura, que representa el cursarla satisfactoriamente, según los planes de estudio vigentes.

Campo numérico entero de 2 dígitos.

créditos actuales

Cantidad de créditos que marca el plan de estudios vigente, que se requieren para acreditar una carrera.

Campo numérico entero de 3 dígitos.

créditos cubiertos

Suma del valor académico de las asignaturas que el alumno ha acreditado para la Facultad de Ingeniería.

Campo numérico entero de 3 dígitos.

créditos cursando

Total de la suma de los créditos que corresponden a las asignaturas que el alumno está cursando en el semestre.

Campo numérico entero de 2 dígitos.

créditos revalidados

Cantidad de créditos reconocidos por la Universidad a un alumno que ha cursado alguna(s) asignatura(s) o carrera en alguna Escuela o Facultad incorporada a la Universidad.

Campo numérico entero de 3 dígitos.

cupo

Número máximo de alumnos que puede tener un grupo inscrito.

Campo numérico entero de 2 dígitos.

delegación/municipio

Para alumnos que radican en el D.F., nombre de la delegación política en que se encuentra su domicilio.

Para alumnos de fuera del D.F., nombre del municipio.

Cadena de 20 caracteres alfabéticos.

dependencia-subdependencia

Dígitos que integran el código programático e indican a la dependencia y subdependencia centralizadora de alguna partida presupuestal.

Véase también "programa-subprograma".

Cadena numérica de 5 caracteres.

digito de control

Son los dos últimos elementos del código programático que distingue a la dependencia centralizadora del gasto e incluye al dígito verificador. Véase también "programa--subprograma".

Cadena numérica de dos caracteres.

domicilio

Lugar en donde reside el alumno. Se forma con la calle, número exterior y número interior de la vivienda.

Cadena de 37 caracteres alfanuméricos.

estado de contratación

Indica si un académico se encuentra actualmente contratado o no. Esto indica que se consideran académicos que alguna vez fueron contratados y que permanecen registrados en el sistema.

Carácter con significado:

C	Si está Contratado,
NULO	No está Contratado.

fecha de actualización alumnos

Año- semestre hasta el que es válida la información de alumnos, al entregar la cinta histórica DGAE.

Campo numérico de tres dígitos, en donde

AAS	
AA	dos últimos dígitos del año.
S	1 ó 2. Semestre non o par.

fecha de adeudo

Fecha a partir de la que se considera que un alumno debe algo, para cada adeudo que él tenga.

Campo tipo fecha.

fecha de alta académicos

Es la fecha de alta del contrato académico vigente, para un miembro del personal académico.

Campo de tipo fecha.

fecha de examen profesional

Fecha en que fue presentado un examen profesional.

fecha de falta

Fecha en que un profesor o académico no asiste a su jornada laboral o impartición de clase.

Campo tipo fecha.

Modelado

fecha de inicio de servicio académico

Sólo para personal académico, representa la fecha en que se dio de alta por primera vez a un miembro del personal académico, con plaza académica.

Campo de tipo fecha.

fecha de la nómina

Fecha en que se generó la nómina cuyos datos se tienen.

Campo tipo fecha.

fecha de nacimiento

Fecha de nacimiento del alumno.

Formato tipo fecha.

fecha límite de contrato académico

Es la fecha en que expira un contrato vigente de personal académico.

Campo tipo fecha.

fecha y hora de extraordinario

Fecha y hora en que se propone efectuar el examen para un grupo extraordinario. Campo tipo fecha.

folio

Número seriado, que el Centro de Cálculo le asigna a cada profesor o personal académico de la Facultad.

Campo numérico de seis dígitos, en donde:

AA NN NV

AA año de registro

NN número seriado, para el año de registro

V dígito verificador.

folio del grupo

Número foliado del acta en que se asienta la calificación de todo el grupo de alumnos que presentan una asignatura.

Cadena de 11 caracteres alfanuméricos.

FFFFF1GGGG en donde:

FFFFFF Folio del grupo.

GGGG Grupo asignado para la calificación.

grupo primer ingreso

Número del grupo que se asignó a un alumno por la DGAE cuando se le dio admisión en la Facultad de Ingeniería y le fue asignado un horario.

Campo numérico de dos dígitos.

grupo tutor

Número asignado por DGAE a cada grupo que participa en el sistema de tutorías. El número se asigna en cada período de inscripciones, por lo que se identifica en forma completa por el año de ingreso a tutoría del alumno.

Campo numérico entero de 2 dígitos.

hora final

Hora en que termina una sesión de una asignatura, para un grupo específico, en un día de la semana.

Campo numérico de 4 caracteres. Rango 0900-2330.

hora final de asesoría

Hora en que termina la sesión de tutoría de un grupo tutoriado.

Cadena de cuatro caracteres numéricos. Rango 0800-2330.

hora inicial

Hora en que comienza la sesión de una asignatura, para un grupo específico, en un día de la semana.

Campo numérico de 4 caracteres. Rango 0700-2200.

hora inicial de asesoría

Hora en que inicia la sesión de tutoría para el grupo.

A partir de esta, y hasta la hora final, los alumnos del grupo podrán visitar a su tutor en el lugar de tutoría.

Cadena de cuatro caracteres numéricos, rango 0700-2200.

horas académicas

Cantidad de Horas/Semana/Mes que especifica el contrato de un miembro del personal académico. El último dígito es la parte decimal (multiplicar por 1).

Campo numérico de 3 dígitos, dos enteros y un decimal.

índice de escolaridad

Se calcula en CEECAFI según fórmula y refleja, en cierta manera, el nivel académico del alumno.

Campo numérico entero con rango 000 - 100.

índice de velocidad

Parámetro utilizado anteriormente por la administración escolar, calculado en el CEECAFI según fórmula.

Campo numérico entero con rango 000 - 100.

inscripciones en ordinario

Total de asignaturas a las que el alumno se ha inscrito durante periodos ordinarios.

Campo numérico entero de 3 dígitos.

**ESTA TERCIA NO DEBE
SALIR DE CUB. BIBLIOTECA**

irregularidad

Código que indica la regularidad del alumno por haberse inscrito, o su irregularidad por no estar inscrito o haber excedido el tiempo para cursar la carrera.

Campo alfabético, con los siguientes valores:

NULO	alumno regular.
S	alumno irregular por no haberse inscrito los últimos 7 semestres.
Q	alumno irregular por haber excedido el tiempo límite de la carrera.
A	alumno irregular por no haberse inscrito los últimos 7 semestres y haber excedido el tiempo límite de la carrera.

laboratorio o práctica

Indica si la asignatura incluye laboratorio o práctica.

Campo alfabético de 1 carácter, con los siguientes valores permitidos:

NULO	sin laboratorio ni práctica
L	con laboratorio
P	con práctica
A	con laboratorio y practica.

lugar de tutoría

Descripción de la ubicación o dirección del lugar en donde se efectúan las sesiones de tutoría, o el tutor recibe a sus alumnos tutoriados.

Cadena de 126 caracteres alfabéticos.

mes de alta administrativos

Representa el mes del año en que se dio de alta a un empleado universitario, con plaza administrativa.

Carácter alfabético con letras de la A a la L, que representan en orden a los meses del año.

motivo de no inscrito

Indica la causa por la que un alumno de la Facultad, ya no se considera que cursa una carrera a la que estuvo inscrito.

Campo alfabético de un carácter, con los valores posibles siguientes:

T	terminó titulado.
C	cambio de carrera.

movimiento

Indica el tipo de movimiento que se desea hacer en trámite de alta, baja o cambio (baja condicionada).

Campo de un carácter, con el siguiente significado:

A	alta
B	baja
C	cambio

nivel

Escalafón o tabulador que rige a una categoría para asignación de sueldo. Cadena de dos caracteres numéricos.

nombre de categoría

Nombre que se le da a la plaza de trabajo, de acuerdo con los reglamentos y normas de la UNAM.

Cadena de 14 caracteres alfabéticos.

nombre de la asignatura

Nombre de una asignatura que se imparte o se ha impartido en la Facultad de Ingeniería.

Campo alfanumérico de 26 caracteres.

nombre de la carrera

Nombre de alguna de las carreras descritas en el atributo clave de la carrera.

Campo alfabético de 34 caracteres.

nombre de la tesis

Título registrado para la tesis que se presenta en un examen profesional. Campo alfanumérico de 64 caracteres.

nombre de lugar de adeudo

Nombre del departamento, laboratorio o biblioteca en que se ofrece servicio de préstamo.

Campo alfabético de 64 caracteres.

nombre de órgano

Nombre de un Órgano de la Facultad de Ingeniería.

Campo Alfabético de 53 caracteres. Sus valores se describen en el atributo clave de órgano.

nombre de procedencia

Nombre de Estado de la República o de alguna nacionalidad extranjera.

Cadena de 21 caracteres alfabéticos.

Modelado

nombre del académico

Nombre y apellidos de un profesor o personal académico de la Facultad de Ingeniería, registrado por CECAFI.
Campo alfabético de 32 caracteres.

nombre del alumno

Nombre con el que un alumno quedó registrado en la UNAM.
El nombre registrado puede contener abreviaciones del nombre real.
Cadena de 32 caracteres alfabéticos.

nombre del departamento

Nombre de un departamento perteneciente a una División de la Facultad de Ingeniería.
Campo alfabético de 32 caracteres.

nombre de día

Cada uno de los días que componen una semana, desde el domingo hasta el sábado.
Campo alfabético de nueve caracteres.

0	domingo
1	lunes
2	martes
3	miércoles
4	jueves
5	viernes
6	sábado

nombre en nómina

Nombre de trabajador académico o administrativo, que está incluido en la nómina de la Facultad de Ingeniería.
Campo alfabético de 25 caracteres.

número de cuenta

Número con el que se identifica a un alumno registrado en la UNAM.
Cadena de 8 caracteres numéricos, en el siguiente formato:

AAFFFFFV

en donde:

AA es el año lectivo en que ingresó a la UNAM.

FFFFF es el número foliado que se le asignó en el registro del período (desde 1971).

V es un dígito verificador.

número de departamento

Número que, para cada División, identifica a cada Departamento de ella. El número es seriado para cada División. Campo numérico entero de dos dígitos.

número de día

Número asignado a cada día de la semana, de acuerdo a la tabla que se presenta en día de la semana. Carácter numérico de un dígito. Rango 0-6.

número de examen

Número serial asignado a cada examen profesional, durante un año. Campo de tipo entero de tres dígitos.

número de grupo

Número que, para cada clave de asignatura en la que se ofrece inscripción, existe en forma seriada, identificando de esta forma a cada grupo ofrecido para la asignatura. Campo numérico entero de dos dígitos.

número de grupo extraordinario

Número que, para cada asignatura de la que se ofrece examen extraordinario, existe en forma seriada, identificando de esta forma a cada grupo. Campo numérico entero de dos dígitos.

número de grupo tutor

Número serial que identifica en forma única a cada grupo tutor que ha sido creado. Campo numérico entero de 4 dígitos.

número de laboratorios inscrito

Para el semestre actual, cantidad de laboratorios en los que está inscrito el alumno en periodo ordinario. Campo numérico entero, con valores entre 0 y 3.

número de movimientos en H.A.

Número de semestres que el alumno ha estado inscrito, tanto en ordinario, como en extraordinario, en la Facultad. Campo numérico entero de tres dígitos.

número de plaza

Número de plaza. Campo numérico de 3 dígitos.

Modelado

número de profesor

Indica si el profesor asignado a un grupo es el primer profesor, segundo profesor, etcétera.
Carácter numérico con los posibles siguientes valores:

0	primer profesor,
1	segundo profesor,
2	tercer profesor,
3	cuarto profesor,
4	quinto profesor.

número de salón

Número que identifica el aula en que se imparte una asignatura, para un grupo determinado.
Campo alfanumérico de 4 caracteres.

número de sorteo

Número asignado semestralmente a los alumnos, en forma aleatoria, para ordenarlos en el proceso de reinscripción.
Campo numérico entero de 5 dígitos.

número de teorías inscrito

Para el semestre actual, cantidad de asignaturas teóricas en las que está inscrito el alumno en periodo ordinario.
Campo numérico entero, con rango entre 0 y 7.

número en nómina

Número asignado por el CECAFI a cada registro de la nómina, con el fin de identificarlo en forma única.
Campo numérico entero de 4 dígitos.

número único de grupo

Número serial asignado a cada grupo para identificarlo en forma única. Con el fin de diferenciar los grupos de semestre non (1) de los de semestre par (2), se asigna una seriación especial a este identificador, de acuerdo a la siguiente regla:

10000 al 14999	asigs. teóricas de semestre 1,
15000 al 19999	laboratorios de semestre 1,
20000 al 24999	asigs. teóricas de semestre 2,
25000 al 29999	laboratorios de semestre 2.

Campo numérico entero de 5 dígitos.

número único de grupo alta

Número del grupo al que se pide una alta en periodo de altas-bajas-cambios. Ver descripción de número único de grupo.

número único de grupo baja

Número del grupo al que se pide una baja en periodo de altas-bajas-cambios. Ver descripción de número único de grupo.

número único de grupo extraordinario

Número serial asignado a cada grupo extraordinario para identificarlo en forma única.
Campo numérico entero de 3 dígitos.

obligatoriedad

Indica si la asignatura se considera como obligada u optativa para el plan de estudios de la carrera correspondiente.

Campo alfabético, con el significado siguiente:

0	obligatoria
5	optativa (selectiva)

opción

Cuando un alumno hace su solicitud de reinscripción, presenta dos alternativas para cada grupo de una asignatura. Al alumno se le asigna el juego de opciones que satisface mas grupos.

Campo numérico, con valores 1 ó 2.

partida

Componente del código programático que clasifica las erogaciones de acuerdo con el objeto específico del gasto. Véase también "programa-subprograma".

Cadena numérica de 3 dígitos.

periodo

Se proponen cuatro periodos para proceso de exámenes extraordinarios, cada semestre.

Campo de un dígito, con posibles valores:

1	primer periodo
2	segundo periodo
3	tercer periodo
4	cuarto periodo

piso a nivel

Atributo booleano que indica si un salón se encuentra a un nivel en el que no es necesario subir o bajar escaleras. Campo de un carácter alfabético:

S Si es necesario subir,
NULO No es necesario subir.

plan escolar asignado

Representa el plan de estudios que seguirá el alumno en el plantel asignado. Para alumnos de la Facultad de Ingeniería, debe ser el mismo. Campo numérico de un dígito.

porcentaje de avance

Valor obtenido al dividir el número de créditos obtenidos por el alumno en su carrera, entre el número total de créditos de su carrera.

Campo numérico comprendido en el rango 0.0 - 100.0.

posible consejero

Indica la posibilidad de un alumno de ser considerado como candidato a consejero.

Campo Alfabético, con las siguientes posibilidades:

NULO alumno NO elegible.
C alumno elegible.

primer salón

Número del salón que se asigna a un grupo de tutoría, para realizar las primeras sesiones.

Campo alfanumérico de 4 caracteres.

primero o segundo

Para cada grupo extraordinario, se propone un primer sinodal y un segundo sinodal para la realización del examen. Campo numérico, con valores:

1 primer sinodal
2 segundo sinodal.

programa-subprograma

Números que identifican el programa y subprograma al que se asigna una partida presupuestal. De acuerdo al Catálogo Presupuestal del Sistema de Presupuesto por programas de la Universidad.

Cadena numérica de 4 dígitos.

NOTA:

Código Programático es un conjunto de dígitos que, ordenados en forma sistemática, se constituye en el elemento central para procesar el cúmulo de información que demanda el sistema de presupuesto por programas.

Ejemplo:

PR	SP	DEP	SD	PAR	DC
13	12	416	01	143	09

De izquierda a derecha, los dígitos representan:

PR	La función y el número del programa.
SP	El subprograma.
DEP	La dependencia.
SD	La subdependencia.
PAR	La partida de gasto.
DC	El dígito de control. De estos dos números, el primero identifica a la dependencia centralizadora del gasto y el segundo constituye el dígito verificador de la correcta integración del código programático.

Para mayor información, consúltese el catálogo presupuestal e instructivo del ejercicio del año correspondiente.

promedio

Valor obtenido de la suma de las calificaciones aprobatorias dividida entre el número de asignaturas aprobadas.

$$\text{Promedio} = \frac{\text{Suma de Calificaciones aprobatorias}}{\text{número de asignaturas aprobadas}}$$

Campo numérico comprendido en el rango 00.00-10.00.

proyección

Campo que indica si un salón está preparado para la proyección de películas y transparencias.

Campo de un carácter:

NULO	no está preparado
S	si está preparado

quincenas pagadas

Representa el número de quincenas en que un trabajador de la Facultad de Ingeniería ha recibido su sueldo.

Campo numérico de 4 dígitos.

registro federal de causante

Registro Federal de Causante del profesor, personal académico o trabajador. Utiliza homónimo.

Campo alfanumérico de 13 caracteres.

ABCDAAAMMDDHHH

A	primer letra del primer apellido
B	primer vocal del primer apellido
C	primer letra del segundo apellido
D	primer letra del primer nombre
AA	año de nacimiento
MM	mes de nacimiento (numérico)
DD	día de nacimiento
HHH	homonimia

En ocasiones, debido a características del nombre, las reglas anteriores cambian de acuerdo a las normas fijadas por la Secretaría de Hacienda.

resultado

Indica si el alumno que realizó el examen profesional, lo aprobó o no.

Campo numérico de un dígito.

1	aprobado
2	aprobado con mención honorífica
3	suspendido

semestre

Semestre en que se sugiere, por la División correspondiente, cursar una asignatura dentro del plan de estudios de la carrera a la que pertenece.

Campo numérico entero, con rango 1-12.

semestre grupo

Indica el semestre para el que un grupo ordinario es vigente.

Campo numérico entero (1 o 2).

semestres sin cursar asignatura

Número de semestres que un alumno lleva sin registrar algún movimiento académico. Campo numérico entero de 3 dígitos.

sexo

Indica el género sexual del alumno.

Carácter alfabético, con el siguiente significado:

M	masculino
F	femenino
NULO	información no definida

sigla

Identificación abreviada de algún Organó de la Facultad de Ingeniería.

Campo alfabético de 6 caracteres (ver clave del Organó).

sueldo

Retribución monetaria que un trabajador de la Universidad recibe a cambio de su trabajo.

Campo numérico entero de 9 dígitos.

teléfono

Número telefónico del alumno.

Cadena de 7 caracteres numéricos.

tipo de alumno

Indica el conjunto de entidades al que pertenece un alumno al que se le asignó un número de sorteo.

Campo de un carácter alfabético, con los siguientes valores posibles:

N	nuevo ingreso,
I	alumno interno,
E	alumno externo.

tipo de clase

Indica el tipo de una clase para un determinado día de la semana y horario.

Carácter alfabético, con los siguientes valores permitidos:

T	clase teórica y
P	clase práctica.

Modelado

tipo de examen

Indica la clase de examen en que un alumno obtiene la calificación registrada.

Carácter numérico, con los siguientes valores:

- | | |
|---|------------------------|
| 0 | examen ordinario, |
| 1 | examen extraordinario. |

tipo de falta

Indica la causa de falta de un profesor o personal académico. Incluye la información de si fue substituido.

Carácter numérico entero de un dígito:

- | | |
|---|--------------------------|
| 1 | falto, |
| 2 | faltó con día económico, |
| 3 | fue substituido. |

título

Grado profesional del académico. En algunos casos se emplea también Señor o Señorita. A pesar de que los valores de este atributo son abreviaturas, en ningún caso se incluye el punto final.

Cadena alfabética de 5 caracteres.

ubicación de edificio

Descripción del lugar donde se encuentra un edificio dentro de las instalaciones de la Facultad de Ingeniería.

Campo alfanumérico de 32 caracteres.

último año-semestre en cursar asignatura

Año (dos últimos dígitos del año) y semestre en que el alumno registró su último movimiento académico en período ordinario.

Cadena de tres caracteres numéricos.

vacantes

Indica el número de lugares disponibles que un grupo tiene para alumnos. Puede existir un número negativo de vacantes.

Campo numérico entero de dos dígitos.

vigencia

Indica si la asignatura se imparte actualmente o no, en la Facultad.

Campo de un carácter, con el siguiente significado:

- | | |
|------|----------------------------|
| NULO | No se imparte actualmente, |
| S | Si se imparte actualmente. |

2.5.4 Tablas ejemplares

Aquí se da la definición de Entidades y Asociaciones de la Base de Datos, mostrando las Tablas que las representan.

Se utiliza la siguiente la convención:

LP	Llave Primaria,
ND	No Duplicados,
NH	No Nulos,
LF	Llave Foránea,
AS	Asignado por el Sistema,
+	Concatenación de atributos,
1,2,...	Comentarios adicionales.

Modelado

Nombre de la Entidad: ACADEMICO

Descripción: Personal de la Facultad, contratado en alguna plaza para actividades académicas por algún departamento de las Divisiones. Pueden existir académicos actualmente no contratados.

ACADEMICO

RFC	TITULO	NOMBRE
LP	NN	NN
REND370529AAA	DR	DANIEL, RESENDIZ NUÑEZ
BAPJ510223ABC	M.C	JUAN, BRANDI PURATA
RODE111111XYZ	ING	J. ENRIQUE, ROBLES DOMINGUEZ

FOLIO	ESTADO
ND	
855549	C
720586	C
770105	

Vease también: GRUPO_TUTOR, CLASE, CONTRATA, FALTA, PAGA, SIJODAL.

FALTA

RFC	NÚMERO ÚNICO	FECHA	TIPO
LP		+	
LF	LF	NN	NN
RODE111111XYZ	10050	30/05/89	1

Nombre de la Entidad: ACTUALIZACION

Descripción: Puesto que la información de la entidad ASIGNATURA y la Asociación "Presentó" se actualiza sólo hasta que DGAE envía a CECAFI la cinta de archivo maestro, en esta entidad se mantiene la fecha (año- semestre) hasta la que es válida esa información. Por otro lado, la fecha de la información de nómina que se tiene aparece también en esta entidad.

ACTUALIZACION

ALUMNOS	NOMINA
ND,NN	ND,NN
891	29/08/89

Modelado

Nombre de la Entidad: ALUMNO

Descripción: Estudiante de la UNAM, que cursa asignaturas en la Facultad de Ingeniería. A un alumno en esta entidad se le ha generado un número de sorteo para su reinscripción semestral.

ALUMNO

CUENTA	SORTEO	CREDITOS	TEORIAS	LABORATORIOS	TIPO
LP	NN,ND				NN
83226100	39	42	S	1	I
83397064	923	42	S	0	I
73031798	5185				N
83057511	5985				N

Véase también: ADEUDA, ABC, INSCRITO, INSCRITO_EXTRA, SOLICITA.

ADEUDA

CUENTA	CLAVE	FECHA
LP		
LF	LF	NN
83226100	1	14/01/88
83397064	2	05/04/89

ABC

CUENTA	NUMERO_UNICO_ALTA	NUMERO_UNICO_BAJA	MOVIMIENTO	OPCION
LP				
LF	LF	LF	NN	
73031798		10001	B	
83057511	10002		A	1
83057511	10031		A	2
83226100	10003	10001	C	1
83226100	10005	10001	C	2

INSCRITO

CUENTA	NUMERO_UNICO
LP	
LF	LF
83057511	10005
73031798	10001

INSCRITO_EXTRA

CUENTA	NUMERO_UNICO
LP	
LF	LF
83397064	1
83057511	5

SOLICITA

CUENTA	NUMERO_UNICO	OPCION
LP		
LF	LF	NN
73031798	10001	1
83057511	10001	2
83057511	10005	1

Modelado

Nombre de la Entidad: ALUMNO_EXTERNO

Descripción: Estudiante de la UNAM, que no está registrado en el plantel 011, y que se inscribe para cursar materias de la Facultad de Ingeniería. Su historia académica no se lleva en esta Facultad.

ALUMNO_EXTERNO

CUENTA	NOMBRE	PROCEDENCIA
LP	NN	NN
82126272	MARIA ELENA CANO SALAZAR	19

Véase también: ALUMNO.

Nombre de la Entidad: ALUMNO_INTERNO

Descripción: Estudiante de la UNAM, registrado en el plantel 011, que: o bien estuvo inscrito en la Facultad de Ingeniería pero no se ha titulado, o está actualmente inscrito en una o varias materias, o estuvo inscrito y ya se tituló. Reconocido por la DGAE.

ALUMNO_INTERNO

CUENTA	NOMBRE
LP	NN
B3226100	HERNANDEZ REYES MARGARITA
B3397064	DIAZ DE SALAS MONTAÑEZ HECTOR
B1161252	PEREZ CASTILLO JOSE FRANCISCO

DOMICILIO	COLONIA
MOCTEZUMA MZ-45 LT-49 SUR 81 NO. 38 ORINOCO 67-312	AJUSCO MERCED BALBUENA SACAHUITZCO

DELEGACION	CP	ESTADO	TELEFONO	NACIMIENTO
COYOACAN	04300	1	6772684	03/04/67
VENUSTIANO CARRANZA		1		12/07/67
BENITO JUAREZ				

NACIONALIDAD	SEXO	INGRESO_ DGAE	INGRESO_ FACULTAD	CREDITOS_ CUBIERTOS
1 F		NN	NN	NN
1	F	86	861	306
1		86	861	272
		84	841	418

CREDITOS_ REVALIDADOS	INSCRIPCIONES	ASIGNATURAS_ ORDINARIO	ASIGNATURAS_ EXTRA
NN	NN	NN	NN
0	37	37	0
0	36	31	1
0	53	51	0

REPROBADAS	PROMEDIO	ESCOLARIDAD	VELOCIDAD
NN	NN	NN	NN
0	9.95	99	105
5	9.00	78	93
2	9.80	94	100

IRREGULARIDAD	CONSEJERO	SIN_CURSAR	ULTIMO_EN_CURSAR
		NN	NN
	C	0	891
	C	0	891
		3	882

ULTIMO_EXTRA	MOVIMIENTOS_HA
	NN
891	37 38 53

Véase también: ESTUDIA, EXAMEN, HA_ESTUDIADO,
PRESENTO, TUTORIA.

ESTUDIA

CUENTA	CLAVE	REGISTRO	AVANCE
LP	+		
LF	LF	NN	NN
83226100	32	861	73.2
83397064	32	861	65.1
81161252	32	841	100.0

EXAMEN

CUENTA	CLAVE	FECHA	NUMERO	RESULTADO
LP	+			
LF	LF	NN	NN	NN
79167505	1739	17/10/89	152	A
79587987	1739	17/10/89	153	A
80242532	1203	20/06/89	98	S

Modelado

HA_ESTUDIADO

CUENTA	CLAVE	REGISTRO	AVANCE	MOTIVO
LF				
LF	LF	NN		NN
79167505	32	821	100	T
83537156	21	841	10	C
79587987	32	791	100	T

PRESENTO

CUENTA	CLAVE	FOLIO	EVALUACION_HA	TIPO	CALIFICACION
LF					
LF	LF	NN,ND	NN	NN	NN
83226100	57	07978991122	861	0	1
83226100	136	11281060004	881	0	1

TUTORIA

CUENTA	NUMERO
LP,LF	LF
73031798	100
83057511	100

Nombre de la Entidad: ALUMNO_NUEVO_INGRESO

Descripción: Estudiante que cursa asignaturas en la Facultad de Ingeniería y del cual aun no se recibe información académica histórica de DGAE por ser de reciente ingreso.

ALUMNO_NUEVO_INGRESO

CUENTA	NOMBRE
LP	NN
73031798	ATA JIMENEZ GILDARDO
83057511	RODRIGUEZ SUAREZ FCO. JAVIER

DOMICILIO	COLONIA
NN	NN
CDA DE CAMINO REAL A STA CECILIA NO 2 COAZINTLA NO 12	SAN ANDRES AHUAYUCAN SAN JERONIMO

DELEGACION	CP	ESTADO	TELEFONO	NACIMIENTO
NN	NN	NN		NN
XOCHIMILCO MAGDALENA CONTRERAS	16810 10400	1 1	5681869	11/03/63 07/08/62

NACIONALIDAD	SEXO	PRIMER_GRUPO	PLAN
LF		NN	NN
9	M	5	1
9	M	27	1

Véase tambien: ALUMNO, PROCEDENCIA.

Modelado

Nombre de la Entidad: ASIGNATURA

Descripción: Asignatura que está o estuvo contemplada en el plan de estudios de alguna carrera de la Facultad. Asignaturas no vigentes pueden tener equivalentes con las vigentes actuales y viceversa. Existen asignaturas que no pertenecen al plan de estudios de alguna carrera, pero que la administración escolar las utiliza.

ASIGNATURA

CLAVE	NOMBRE	CREDITOS	LABORATORIO
LP	NN	NN	
138	DISPOSITIVOS ELECTRONICOS	10	L
57	COMPUTADORAS Y PROGRAMACION	?	
75	PROGRAM. LOGICA Y CAR. LENGUJES	8	

INICIO	VIGENCIA	DEPARTAMENTO
		LF
701	S	5
792	S	21
801	S	6

Véase también: GRUPO, GRUPO_EXTRA, COMPONE, EQUIVALENCIA, PRESENTO, SERIACION.

EQUIVALENCIA

ANTIGUA	ACTUAL
LP	+
LF	LF
61	57

SERIACION

ANTECEDENTE	CONSECUENTE
LP	
LF	LF
57	75

Modelado

Nombre de la Entidad: CARRERA

Descripción: Area profesional de Licenciatura, que se ofrece en esta Facultad. Se coordina por una división.

CARRERA

CLAVE	NOMBRE	ORGANO	CREDITOS
LP	NN, ND	LF	NN
32	INGENIERO EN COMPUTACION	3	391
21	INGENIERO CIVIL	2	403

Véase también: COMPONE, ESTUDIA, HA_ESTUDIADO.

COMPONE

CARRERA	ASIGNATURA	SEMESTRE	OBLIGATORIEDAD
LP			
LF	LF	NN	
32	138	5	0
29	138	5	0

Nombre de la Entidad: DEPARTAMENTO

Descripción: Elemento que pertenece al organigrama de un Organo, en el nivel bajo la Jefatura. Se encarga de coordinar asignaturas y académicos a su cargo.

DEPARTAMENTO

CLAVE	ORGANO	NUMERO	NOMBRE
LP,AS	LF	NN	NN
5	3	2	DEPARTAMENTO DE ELECTRONICA
6	3	5	DEPARTAMENTO DE COMPUTACION

Véase también: ASIGNATURA, LUGAR_DE_ADEUDO, CONTRATA.

CONTRATA

CLAVE	RFC
LP	
LF	LF
6	REND370529AAA
5	RODE111111XYZ

Modelado

Nombre de la Entidad: DÍA

Descripción: Día de la semana: lunes, martes, miércoles, jueves, viernes, sábado, domingo.

DÍA

NUMERO	NOMBRE
LP	NN,ND
0	DOMINGO
1	LUNES

Véase también: CITA, CLASE.

Nombre de la Entidad: EDIFICIO

Descripción: Lugar dentro de las instalaciones de la Facultad de Ingeniería en donde se encuentran grupos de salones.

EDIFICIO

CLAVE	UBICACION
LP,AS	NN
1	ALA DESTE DE CIENCIAS BASICAS
2	EDIFICIO DE LA DIME

Véase también: SALON.

Modelado

Nombre de la Entidad: GRUPO

Descripción: Opción de una asignatura, con uno o dos profesores (ACADEMICO) a su cargo, dentro de un horario, en el cual los alumnos se inscriben para cursar una materia.

GRUPO

NUMERO_UNICO	NUMERO	SEMESTRE	CUPO	VACANTES	ASIGNATURA
LP, AS	NN	NN	NN	NN	LF
10050	1	1	60	60	138
10150	7	1	20	20	213

Véase también: ABC, CLASE, FALTA, INSCRITO, SOLICITA.

CLASE

NUMERO_UNICO	RF	DIA	SALON	TIPO
LP	+	+	+	+
LF	LF	LF	LF	
10050	RODE111111XYZ	2	304A	T
10050	RODE111111XYZ	4	304A	T
10103	BAPJ510223ABC	2	116	T
10103	GOCJ651202EXN	2	116	P
10103	BAPJ510223ABC	4	116	T
10103	GOCJ651202EXN	4	116	P

NUMERO_PROFESOR	TIPO	HORA_INICIAL	HORA_FINAL
NN	NN	NN	NN
0	T	700	900
0	T	700	900
0	T	900	1100
1	T	900	1100
0	P	900	1100
1	T	900	1100

Nombre de la Entidad: GRUPO_EXTRA

Descripción: Grupo que es abierto para inscripción a exámenes extraordinarios. Los exámenes en estos grupos se realizan en alguno de los cuatro periodos de fechas en un semestre.

GRUPO_EXTRA

NUMERO_UNICO	NUMERO	PERIODO	FECHA	SALON	ASIGNATURA
LP,AS	NN	NN	1	LF	LF
1	51	2	13/05/89 11:30	L 02	138
2	51	2	05/05/89 17:00	304A	75

1 - Se considera también la hora, además de la fecha.

Véase también: INSCRITO_EXTRA, SINODAL.

SINODAL

NUMERO_UNICO	RFC	PRIMERO_SEGUNDO
LP,AS	+	
LF	LF	NN
1	REND370529AAA	1
2	BAPJ51022JABC	2

Modelado

Nombre de la Entidad: GRUPO_TUTOR

Descripción: Organización que agrupa a un conjunto de alumnos que tienen a un mismo profesor de la Facultad de Ingeniería como tutor.

GRUPO_TUTOR

NUMERO	INICIO	GRUPO	PRIMER_SALON	LUGAR	RFC
LP,AS	NN	NN	NN	1	LF
100	89	1	002A	DIRECCION PRI...	REND370529AAA
57	88	38	004B	PISO CUATRO ...	BAPJ510223ABC

1 - Tiene longitud de 126 caracteres.

Véase también: CITA, TUTORIA.

CITA

NUMERO_GT	DIA	HORA_INICIAL	HORA_FINAL
LP	+		
LF	LF	NN	NN
100	3	1730	2100
57	2	1800	1900
57	4	1800	1900

Nombre de la Entidad: LUGAR_DE_ADEUDO

Descripción: Departamento, laboratorio, biblioteca, o lugar en que se da un servicio de préstamo de material, y en el que un alumno se puede encontrar como deudor de material.

LUGAR_DE_ADEUDO

CLAVE	NOMBRE	DEPARTAMENTO
LP, AS	NN, ND, 1	LF
1	LABORATORIO DE ELECTRONICA	5
2	SALA DE TERMINALES	6

1 - Tiene longitud de 64 caracteres.

Véase también: ADEUDA.

Modelado

Nombre de la Entidad: NOMINA

Descripción: Es el conjunto de registros de los trabajadores académicos y administrativos que pertenecen a la Facultad de Ingeniería. La información se recibe periódicamente de la Dirección General de Servicios de Cómputo Administrativo.

NOMINA

NUMERO	RFC	NOMBRE
LP,AS	NN	NN
11	LOML451031	LOZANO MONTES DE O LUCILA
12	LOML451031	LOZANO MONTES DE O LUCILA

NOMBRE_CATEGORIA	CLAVE_CATEGORIA	NIVEL	PLAZA	SUELDO
NN		NN	NN	NN
JEFE DE SECCION	A05	50	001	123456
MEDIA PLAZA	A05	50	001	23456

PROGRAMA	DEPENDENCIA	PARTIDA	CONTROL	SERVICIO	INICIO_SERVICIO
NN	NN	NN	NN		
1301	41601	151	00	0	00/00/00
1301	41601	155	02	0	00/00/00

PAGADAS	MES_ALTA_ADM	ANO_ALTA_ADM	ALTA_ACAD	LIMITE
NH				
54B	L	65	16/12/87	00/00/00
54B	L	65	16/12/87	00/00/00

DESCONOCIDA	HORAS
003	00.0
003	00.0

Véase también: PAGA.

PAGA

NUMERO	RFC
LP,LF	LF
11	LOML451031
12	LOML451031

Modelado

Hombre de la Entidad: ORGANO

Descripción: Elemento de la organización de alto nivel de la Facultad de Ingeniería. Algunos órganos son las Divisiones, responsables de organizar y administrar la estructura, planes de estudio, materias y personal académico de las carreras profesionales de licenciatura. También se incluyen Secretarías, Centros y otros.

ORGANO

CLAVE	NOMBRE
LP, AC	NN, ND
0	DIRECCION
1	SECRETARIA GENERAL
2	DIVISION DE INGENIERIA CIVIL, TOPOGRAFICA Y GEODESICA
3	DIVISION DE INGENIERIA MECANICA Y ELECTRICA

SIGLA
ND
SG DICTG DIME

Véase también: CARRERA, DEPARTAMENTO, SALON.

Nombre de la Entidad: PROCEDENCIA

Descripción: Nombre del Estado de procedencia, para alumnos nacionales, o nacionalidad, para alumnos extranjeros, de la Facultad.

PROCEDENCIA

CLAVE	NOMBRE
LP	NN,ND
26	SONORA
17	MORELOS
5	CHIAPAS
9	DISTRITO FEDERAL
40	AUSTRALIANA

Véase también: ALUMNO_INTERNO, ALUMNO_NUEVO_INGRESO.

Modelado

Nombre de la Entidad: SALON

Descripción: Aula o laboratorio en que se imparte una asignatura de la Facultad de Ingeniería. Cada salón cuenta con un grupo de características, como el ser de proyección, lugar, etcétera.

SALON

NUMERO	ORGANO	PROYECCION	CAPACIDAD	EDIFICIO	PISO
LP	LF		NN	LF	1
L-02	3	S	50	1	S
304A	2		20	3	

1 - Valor booleano.

Véase también: GRUPO_EXTRA, CLASE.

Nombre de la Entidad: TESIS

Descripción: Trabajo escrito que un alumno presenta como requisito para la obtención de su título profesional. Por reglamento, este trabajo escrito se debe defender en un EXAMEN profesional oral.

TESIS

CLAVE	NOMBRE
LP,AS	NN,1
1739	DISEÑO DE BASE DE DATOS PARA EL CENTRO DE CÁLCULO ...
1203	DISCUSIÓN SOBRE EL COMPORTAMIENTO DE MATERIALES

1 - tiene longitud de 64 caracteres.

Véase también: EXAMEN.

Para formar la primera tabla se tomó cada Entidad o Asociación de los títulos superiores y se marcó con una X en la columna vertical el cuadro que la intersecta con el nombre de la Entidad de la cuál depende para existir. La dependencia se decide observando el rectángulo o rombo que la representa en el DEA y siguiendo las líneas de conexión: cualquier Asociación depende de las Entidades que vincula, mientras que el grado de asociación y de pertenencia delata a las Entidades debiles.

La segunda tabla se obtiene a partir de la primera. El primer paso es colocar el numero 1 sobre las Entidades que no presentan una cruz a lo largo de la columna vertical. Después, se substituyón por unos todas las cruces que aparezcan en la línea horizontal de cada Entidad numerada con uno. El siguiente paso es colocar un dos sobre las Entidades o Asociaciones a las que haya aparecido un uno en su línea vertical, para después substituir por el número dos a las cruces que se encuentren en la línea horizontal de las Entidades identificadas con ese número.

Esos pasos se repiten hasta que todas las columnas tengan un número de precedencia, incrementando el número que se va escribiendo. En caso de encontrarse varios números en una columna vertical, se tomará el de mayor valor como el número de esa columna.

Los números obtenidos en la parte superior de la tabla fueron de gran utilidad, ya que determinaron el orden de carga de información en la Base de Datos.

2.5.6 Tabla de accesos

Las siguientes tablas muestran el tipo de acceso que hace cada Coordinación del Centro de Cálculo sobre los atributos de la Base de Datos. Se sigue la convención descrita a continuación:

- A Alta del atributo. Por consiguiente, la creación del renglón o tuplo completo.
- B Baja del atributo. Lo que implica la eliminación completa del renglón.
- C Cambio del atributo. Que es el modificar su valor, sin afectar al resto del renglón.
- D Consulta del atributo. Lectura del valor del atributo, sin alterar en ningún momento su valor.

Estos permisos son dinámicos, y pueden ser modificados de acuerdo con las necesidades y características de los procesos de cada Coordinación.

Modelado

Tabla Atributo	Coordinación	Acadé- mica	Adminis- trativa	Anexo	Investi- gación	Planea- ción	Secreta- ría
ACADEMICO RFC TITULO NOMBRE FOLIO ESTADO		D D D	D D				ABCD ABCD ABCD ABCD
ACTUALIZACION ALUMNOS NOMINA		CD	D		CD		
ALUMNO CUENTA SORTEO CREDITOS TEORIAS LABORATORIOS TIPO			D CD CD CD		ABCD ABCD ABCD ABCD ABCD		
ALUMNO_EXTERNO CUENTA NOMBRE PROCEDENCIA					ABCD ABCD ABCD		

* Quién en realidad da de alta un renglón de esta Tabla, es el Administrador de la Base de Datos.

Tabla Atributo	Coordinación	Acadé- mica	Adminis- trativa	Anexo	Investi- gación	Planea- ción	Secre- taría
ALUMNO_INTERNO							
CUENTA			D		ABCD	D	D
NOMBRE			D		ABCD	D	D
DOMICILIO			CD		AB D		
COLONIA			CD		AB D		
DELEGACION			CD		AB D		
CP			CD		AB D		
ESTADO			CD		AB D		
TELEFONO			CU		AB		
NACIMIENTO			CD		AB		
NACIONALIDAD					AB		
SEXO			CD		ABCD		
INGRESO_DEGAE					ABCD		
INGRESO_FACULTAD					ABCD		
CREDITOS_CUBIERTOS					ABCD		
CREDITOS_REVALUADOS					ABCD		
INSCRIPCIONES					ABCD		
ASIGNATURAS_ORDINARIO					ABCD		
ASIGNATURAS_EXTRA					ABCD		
REPROBADAS					ABCD		
PROMEDIO					ABCD		
ESCOLARIDAD					ABCD		
VELOCIDAD					ABCD		
IRREGULARIDAD					ABCD		
CONSEJERO					ABCD		
SIN_CURSAR					ABCD		
ULTIMO_EN_CURSAR					ABCD		
ULTIMO_EXTRA					ABCD		
MOVIMIENTOS_HA					ABCD		
ALUMNO_NUEVO_INGRESO							
CUENTA			ABCD		D		
NOMBRE			ABCD				
DOMICILIO			ABCD				
COLONIA			ABCD				
DELEGACION			ABCD				
CP			ABCD				
ESTADO			ABCD				
TELEFONO			ABCD				
NACIMIENTO			ABCD				
NACIONALIDAD			ABCD				
SEXO			ABCD				
PRIMER_GRUPO			ABCD				
PLAN			ABCD				

Modelado

Tabla	Coordinación	Acadé- mica	Adminis- trativa	Anexo	Investi- gación	Planea- ción	Secre- taría
ASIGNATURA							
CLAVE			D		D	D	ABCD
NOMBRE			D		D	D	ABCD
CREDITOS							ABCD
LABORATORIOS							ABCD
INICIO							ABCD
VIGENCIA						D	ABCD
DEPARTAMENTO						D	ABCD
CARRERA							
CLAVE					D		ABCD
NOMBRE					D		ABCD
ORGANO					D		ABCD
CREDITOS							ABCD
DEPARTAMENTO							
CLAVE			D		D	D	ABCD
ORGANO			D		D	D	ABCD
NUMERO			D		D	D	ABCD
NOMBRE			D		D	D	ABCD
DIA							
NUMERO			D				D
NOMBRE			D				D
EDIFICIO							
CLAVE							ABCD
UBICACION							ABCD
GRUPO							
NUMERO_UNICO			D				ABCD
NUMERO			D				ABCD
SEMESTRE			D				ABCD
CUPO			D				ABCD
VACANTES			D				ABCD
ASIGNATURA			D				ABCD
GRUPO_EXTRA							
NUMERO_UNICO			ABCD				
NUMERO			ABCD				
PERIODO			ABCD				
FECHA			ABCD				
SALON			ABCD				
ASIGNATURA			ABCD				

* Quién en realidad da de alta un renglón de esta tabla, es el Administrador de la Base de Datos.

Tabla	Coordinación	Acadé- mica	Adminis- trativa	Anexo	Investi- gación	Planea- ción	Secreta- ría
GRUPO_TUTOR							
NUMERO			ABCD				
INICIO			ABCD				
GRUPO			ABCD				
PRIMER_SALON			ABCD				
LUGAR			ABCD				
RFC			ABCD				
LUGAR_DE_ADEUDO							
CLAVE			ABCD				
NOMBRE			ABCD				
DEPARTAMENTO			ABCD				
NOMINA							
NUMERO		ABCD					
RFC		ABCD					
NOMBRE		ABCD					
NOMBRE_CATEGORIA		ABCD					
CLAVE_CATEGORIA		ABCD					
NIVEL		ABCD					
PLAZA		ABCD					
SUELDO		ABCD					
PROGRAMA		ABCD					
DEPENDENCIA		ABCD					
PARTIDA		ABCD					
CONTROL		ABCD					
SERVICIO		ABCD					
INICIO_SERVICIO		ABCD					
PAGADAS		ABCD					
MES_ALTA_ADM		ABCD					
ANO_ALTA_ADM		ABCD					
ALTA_ACAD		ABCD					
LIMITE		ABCD					
DESCONOCIDA		ABCD					
HORAS		ABCD					
ORGANO							
CLAVE			D		D	D	ABCD
NOMBRE			D		D	D	ABCD
PROCEDENCIA							
CLAVE			ABCD				
NOMBRE			ABCD				

Tabla atributo	Coordinación	Acadé- mico	Adminis- trativa	Anexo	Investi- gación	Planea- ción	Secre- taría
SALON NUMERO ORGANO PROYECCION CUPO EDIFICIO PISO			D				ABCD ABCD ABCD ABCD ABCD ABCD
TESIS CLAVE NOMBRE							ABCD ABCD
ADEUDA CUENTA CLAVE FECHA			ABCD ABCD ABCD				
ABC CUENTA NUMERO_UNICO_ALTA NUMERO_UNICO_BAJA MOVIMIENTO OPCION			ABCD ABCD ABCD ABCD				
CITA NUMERO_GT DIA HORA_INICIAL HORA_FINAL			ABCD ABCD ABCD ABCD				
CLASE NUMERO_UNICO RFC DIA SALON NUMERO_PROFESOR TIPO HORA_INICIAL HORA_FINAL			D D D D D D D				ABCD ABCD ABCD ABCD ABCD ABCD ABCD
COMPONE CARRERA ASIGNATURA SEMESTRE OBLIGATORIEDAD					D D		ABCD ABCD ABCD ABCD

Tabla Atributo	Coordinación	Acadé- mica	Adminis- trativa	Anexo	Investi- gación	Planea- ción	Secreta- ría
CONTRATA CLAVE RFC							ABCD ABCD
EQUIVALENCIA ANTERIOR ACTUAL							ABCD ABCD
ESTUDIA CUENTA CLAVE REGISTRO AVANCE			ABCD ABCD ABCD AB		B B B BCD		
EXAMEN CUENTA CLAVE FECHA NUMERO RESULTADO							ABCD ABCD ABCD ABCD ABCD
FALTA RFC NUMERO_UNICO FECHA TIPO							ABCD ABCD ABCD ABCD
HA_ESTUDIADO CUENTA CLAVE REGISTRO AVANCE MOTIVO			ABCD ABCD ABCD ABCD ABCD		A A A A A		D D D
INSCRITO CUENTA NUMERO_UNICO			ABCD ABCD				
INSCRITO_EXTRA CUENTA NUMERO_UNICO			ABCD ABCD				
PAGA NUMERO RFC		ABCD ABCD					

Modelado

Tabla Atributo	Coordinación	Acadé- mica	Adminis- trativa	Anexo	Investi- gación	Planca- ción	Secre- taria
PRESENTO CUENTA CLAVE FOLIO EVALUACION_HA TIPO CALIFICACION					ABCD ABCD ABCD ABCD ABCD	0 D 0 0 0	
SERIACION ANTECEDE PRECEDE							ABCD ABCD
SINODAL NUMERO_UNICO RFC PRIMERO_SEGUNDO			ABCD ABCD ABCD				
SOLICITA CUENTA NUMERO_UNICO OPCION			ABCD ABCD ABCD				
TUTORIA CUENTA NUMERO			ABCD ABCD				

Selección de un Manejador de Bases de Datos

Una vez realizado el Modelo lógico, el siguiente paso consistió en seleccionar un manejador de bases de datos comercial, que permitiera implantarlo y explotarlo de la mejor manera posible.

Para lograr esto, se plantearon diferentes alternativas que fueron estudiadas en forma paralela considerando las principales características que debe tener un Sistema Manejador de Bases de Datos (SMBD). Se evaluó cada una de ellas mediante pruebas de desempeño especialmente diseñadas para el efecto y finalmente se hizo la selección en base a los resultados obtenidos. En este capítulo se muestran cada una de las etapas de este proceso.

3.1 Alternativas

En la primera fase de este estudio, se pensó en varias alternativas, pues son muchos los manejadores de bases de datos existentes en el mercado. Sin embargo, al primer análisis las alternativas se redujeron considerablemente. La Facultad de Ingeniería había recibido recientemente como donación dos manejadores de bases de datos de calidad comprobada en el

mercado: ADABAS y ORACLE. Este hecho, así como la premisa de la escasez de recursos económicos de la Facultad marcó la pauta del estudio. La Facultad de Ingeniería contaba con dos buenos manejadores de bases de datos, ambos asignados al Centro de Cálculo y funcionando en el sistema VAX. La selección sería sobre alguno de ellos.

3.2 Estudio teórico

Para la realización de este estudio, previamente se determinaron los puntos que a nuestro criterio son los más importantes para el análisis de un manejador de bases de datos. Estos son los siguientes:

Características generales del SMBD

- Antecedentes.
- Administración del espacio en disco.
- Representación y definición de datos.
- Métodos de acceso y recuperación.
- Control de acceso múltiple.
- Límites.

Características y facilidades para el usuario

- Facilidades de consulta.
- Facilidades de actualización.
- Diccionario de datos.
- Interacción con otros lenguajes.
- Productos adicionales.

Características de administración

- Seguridad.
- Respaldo y recuperación de errores.
- Facilidades de monitoreo.
- Utilerías para administración en general.

Facilidades que ofrece el proveedor

- Soporte.
- Calidad de la documentación disponible.

Requerimientos del SMBD

- Cantidad de espacio en disco.
- Cantidad de memoria principal.

A continuación se menciona el comportamiento y características tanto de ORACLE como de ADABAS respecto a los puntos anteriores.

3.2.1 Características generales del SMBD

3.2.1.1 Antecedentes

ADABAS

Es considerada como una base de datos cuasi-relacional, que aunque no está conformada por tablas, provee elementos básicos para realizar un procesamiento relacional a través de listas invertidas.

Este producto es distribuido por SOFTWARE A.G. Surgió en 1971 en un ambiente IBM y actualmente cuenta con más de 3000 instalaciones en el mundo. Funciona en ambientes MVS, VM/CMS y

DOS/VSE de IBM, así como en VAX/VMS de DIGITAL y computadoras Siemens.

ORACLE

Es una base de datos de tipo relacional distribuida por la compañía ORACLE CORPORATION, la cual surgió en 1979 utilizando el trabajo de Ted Cood para desarrollar un SMDR relacional que fuera usado en una gran variedad de computadoras. Actualmente cuenta con una gran portabilidad debido a que está escrito en el lenguaje de programación "C".

Este manejador fue originalmente implantado en un sistema VAX/VMS. Actualmente está funcionando también en otros ambientes como son: VM/CMS y MVS de IBM, UNIX y MSDOS, teniendo más de 4000 productos instalados.

3.2.1.2 Administración del espacio en disco

ADABAS

El modelo que emplea ADABAS para manejar el espacio en disco separa totalmente los datos de las estructuras de acceso a ellos. Consiste de las siguientes áreas:

- Almacenamiento de datos (DATA STORAGE).
- Asociador (ASSOCIATOR).
- Área de trabajo (WORK).
- Área de ordenamiento (SORT).
- Área temporal (TEMP).

En el DATA STORAGE se encuentran en forma comprimida todos los datos de la base. Esta área es independiente de las estructuras de acceso a ella y permite hacer una definición simple e independiente de los archivos creados. Debido a esto es posible agregar nuevos archivos para nuevas aplicaciones sin reorganizar la información.

Esta área se encuentra dividida en bloques, y estos son de longitud fija. Dentro de los bloques se reserva un área de sobreflujo o "padding" especial para el crecimiento de los registros dentro del bloque. Los registros en la base de datos son almacenados dentro del bloque como una cadena de longitud variable comprimida.

Si existe un sobreflujo, el registro es exportado hacia otro bloque, dejando su espacio libre para ser agregado al área de sobreflujo. Este espacio es inmediata y totalmente reutilizable. La cantidad de área de sobreflujo es establecida archivo por archivo por medio de la utilería ADALOD.

También existe la posibilidad de disminuir o agregar espacio a las áreas asignadas al ASSOCIATOR y al DATA STORAGE y esto se hace a través de la utilería ADADBM.

El ASSOCIATOR es una área en disco que contiene las estructuras necesarias para tener acceso a los registros en forma eficiente. Estas estructuras son las siguientes:

- Listas invertidas.
- Convertidor de direcciones.

Su mantenimiento es totalmente automático y transparente para el usuario. El convertidor de direcciones traslada las direcciones lógicas de los registros en direcciones físicas. Las listas invertidas son diseñadas de acuerdo a un descriptor o llave con el cual se pueda localizar eficientemente la información en la base datos, existiendo una lista invertida por cada descriptor que se defina en ésta.

El área de WORK es utilizada para almacenar listas temporales, cadenas usadas durante búsquedas complejas e información concerniente a la protección de los datos.

Las áreas de SORT y TEMP se emplean al momento de crear los descriptores para un archivo. Una vez que se generan todos los descriptores necesarios para una aplicación se puede prescindir de estas dos áreas y por lo tanto pueden ser borradas.

Es necesario mencionar que en ADABAS el dispositivo de almacenamiento debe estar organizado en una forma específica. Se deben emplear dos utilerías para preparar el área a ser utilizada. A través de ADADEV se aparta un conjunto de cilindros que equivalen a un grupo de bloques en VMS. En la utilería se le indica el número de cilindros a ser apartados. Posteriormente se usa ADAFRM para crear y dar formato a las áreas de la base de datos (ASSU, DATA, WORK, TEMP y SORT).

Para disminuir el espacio en disco ADABAS realiza compresión a nivel de campo en cada uno de los registros. Los tipos de compresión son las siguientes:

- # Compresión ordinaria.
- # Compresión fija.
- % Supresión de valores nulos.

La compresión ordinaria elimina todos los blancos finales en un campo alfanumérico, los ceros iniciales en un campo numérico y agrega el tamaño del registro en bytes al inicio del campo.

La compresión fija consiste en almacenar el campo en su forma estándar; ésta no agrega la longitud en bytes.

La compresión de valores nulos elimina todos los campos que toman un valor nulo, agregando un indicador de campo vacío del tamaño de un byte (en la compresión ordinaria se indicaría con 2 bytes, uno indicando la longitud y otro el valor que se repite). El indicador contiene el número de campos nulos y el valor indicador de nulo. Si el campo no es nulo se aplica la compresión ordinaria.

La compresión de valor nulo para campos de valor múltiple y grupos periódicos reduce en gran medida la cantidad de espacio en disco, así como los requerimientos de procesamiento interno en estos tipos de campo.

La compresión de los datos se realiza mediante la utilidad ADACMP, indicándole el tipo de compresión deseada al momento de cargar los datos a la base.

Si se desea descomprimir la información para una aplicación (exportar datos) primero se descarga la información de la base

con la utilería ADAULD y posteriormente se descomprime con la utilería ADADCU.

ORACLE

La base de datos consiste de al menos una unidad lógica llamada partición. Esta partición esta constituida por al menos un archivo físico.

La partición es el área principal de almacenamiento de la base de datos. Pueden existir una o varias particiones. La información que se almacena en esta area es:

- Tablas del diccionario de datos.
- Tablas temporales.
- Tablas de ayuda en línea.
- Tablas del usuario.

Esta partición está constituida por bloques contiguos. Los bloques contiguos se obtienen ejecutando la utileria CCF (Create Contiguos File) de ORACLE, la cual además de apartar localidades contiguas de espacio en disco efectúa una limpieza de estas localidades.

Cuando se termina el espacio disponible en la base de datos, es decir, cuando se ha agotado una partición, se pueden sumar nuevos archivos a la base de datos o agregar mas particiones. Para agregar espacio a la partición se ejecuta la instrucción ALTER PARTITION de SQL¹.

Si se desea observar la cantidad de espacio disponible en la partición, se consulta la tabla SYSS\$SPACEMAP la cuál

¹ SQL es el lenguaje de Definición y Manipulación de datos empleado por ORACLE.

contiene el bloque inicial y el bloque final de cada uno de sus espacios contiguos.

En ORACLE la información se encuentra organizada en tablas, éstas son la unidad básica donde almacena toda la información. Para crear una tabla es necesario tener en la partición un número inicial de bloques contiguos libres, en caso contrario la tabla no puede ser creada.

Cuando una tabla es creada, se reserva automáticamente espacio para su futuro crecimiento. Existe una manera de controlar el crecimiento de las tablas, definiendo el conjunto de características que tendrá: número de bloques inicialmente apartados, número máximo de veces que podrá incrementarse y número de bloques apartados en cada incremento (esto se conoce como definición de espacio).

Cuando la tabla excede el número máximo de veces que puede incrementarse, el usuario simplemente debe crear otra tabla con una definición de espacio diferente (obviamente indicando en esta nueva definición un número máximo de veces mayor en que podrá incrementarse) y copiar la tabla anterior a la tabla actual.

La reorganización del espacio que ocupa la base de datos no es transparente para el administrador. En ocasiones se pueden tener muchos bloques libres en la partición pero no contiguos, por lo que no pueden crearse nuevas tablas. Para conseguir suficientes bloques libres contiguos es necesario exportar totalmente la base de datos e importarla, reorganizando así el espacio ocupado.

Sólo cuando una tabla es borrada totalmente, el espacio que ocupaba queda libre para ser usado por otras tablas. Cuando la tabla desocupa una parte de los bloques asignados a ella (eliminación de renglones o índices), este espacio retorna como libre a ser reutilizado por la misma tabla.

Para disminuir el espacio ocupado, ORACLE almacena los datos en cadenas de longitud variable, eliminando blancos y ceros no significativos. Los valores nulos no ocupan espacio. Para el almacenamiento de los índices cuenta con dos tipos de compresión:

- Compresión "forward".
- Compresión "rear".

Ambos tipos de compresión se aplican al mismo tiempo obteniéndose la parte del índice que hace que este valor sea único tanto entre los valores que están inmediatamente antes o inmediatamente después. A través de este tipo de compresión se reduce el almacenamiento en disco y por lo tanto se reduce el tiempo de entrada/salida.

3.2.1.3 Representación y definición de datos.

ADABAS

En ADABAS la definición de datos se hace en el momento de cargar un archivo. Para cargar un archivo es necesario procesarlo mediante dos utilerías: ADACMP y ADALDD.

ADACMP comprime la información, de forma que pueda ser utilizada por la utilería que carga el archivo (ADALDD). Con

esta misma utilidad, se define la estructura lógica del archivo, almacenándose en una tabla llamada FDT (Field Definition Table).

A continuación se mencionan las características que pueden ser definidas a un campo.

Nivel. Es un número utilizado para agregar campos. Normalmente es 1. Si es mayor indica que es parte del campo inmediato anterior que tiene un número menor.

Nombre. Está limitado a dos caracteres. El primero debe ser alfabético. El segundo puede ser alfabético o numérico. No permite letras minúsculas ni palabras reservadas.

Longitud. Sirve para indicar a la utilidad la longitud en el campo, tal como aparece en el registro de entrada y la longitud en que se almacenará en la FDT. La máxima longitud depende del tipo de dato, si ésta no se especifica, se considera de longitud variable.

Formato. Los tipos de formato permitidos para almacenar los datos son:

- A - Alfabético.
- B - Binario.
- F - De punto fijo.
- P - Decimal empacado.
- U - Decimal no empacado.
- G - De punto flotante.

Otras características. Se pueden definir características adicionales como las siguientes:

- # DE (Descriptor). Indica que el campo será un descriptor, es decir, se le generará una lista invertida para que sea posible usarla como llave en operaciones de búsqueda.
- # UQ (Descriptor Único). Indica que el descriptor debe contener valores distintos para cada registro del archivo. FI (Almacenamiento fijo). No se realiza compactación en el campo.
- # NU (Supresión de valores nulos). Se indica que si el campo tiene valor nulo, sólo almacenará un byte indicándolo. En caso de haber campos nulos consecutivos, sólo almacenará el número de campos.
- # MU (Campo de valor múltiple). Indica que un campo contendrá varios valores posibles. En primer lugar se almacena un contador de cuántos valores existen y posteriormente cada uno de ellos. Este contador se actualiza automáticamente. Puede haber hasta 191 valores.
- # PE (Grupo periódico). Indica la existencia de un grupo periódico. Grupos periódicos son un conjunto de uno o más campos que pueden ocurrir hasta 99 veces. El campo denominado como PE debe tener nivel 1 y cada uno de sus subcampos niveles subsecuentes. Dentro de cada uno de estos subcampos puede haber descriptors o campos de valor múltiple. Si el número de ocurrencias no es constante para todos los registros de entrada, en cada uno se debe indicar en 1 byte el número de ocurrencias, inmediatamente antes de la primera.

Es posible definir descriptores compuestos por una parte de un campo (subdescriptor) o compuestos por varios campos o parte de ellos (superdescriptor). También es posible definir descriptores fonéticos, los cuales son usados para operaciones de búsqueda basadas en su mismo valor fonético.

ORACLE

En ORACLE la definición de datos se hace a través de SQL. SQL contiene comandos para definir tablas (CREATE TABLE), para modificar su estructura (ALTER TABLE) o para eliminarlas (DROP TABLE). En general, estos comandos son de fácil uso.

Mediante el comando CREATE TABLE se definen los nombres, longitud y tipo de cada columna. Opcionalmente se puede indicar si los registros serán cargados desde otra tabla. Los tipos de datos que maneja se describen a continuación.

CHAR. Puede almacenar cualquier carácter alfanumérico. Estos son almacenados en cadenas de longitud variable, teniendo un máximo de 240. Los blancos al final no son almacenados, a menos que explícitamente se requiera.

NUMBER. Puede almacenar números tanto enteros como reales. Los números mas grandes que puede almacenar son del orden de 10 elevado a la 65. Alcanza una precisión de 38 dígitos. Estos números son almacenados en formato de longitud variable, pudiendo ocupar hasta 20 bytes, el primero de éstos es ocupado para almacenar el signo y exponente.

LONG. Sirve para almacenar cadenas de caracteres de longitud variable conteniendo hasta 65535 caracteres. Se usa para almacenar documentos cortos. Este tipo tiene algunas restric-

ciones, entre las cuales se puede mencionar que no se permite más de una columna de este tipo en la misma tabla. Una columna de este tipo no se puede indexar ni la tabla se puede agrupar. Además una columna definida así, no puede aparecer en cláusulas como WHERE, GROUP BY, ORDER BY y otras.

DATE. Este tipo de dato puede almacenar fechas en un formato de longitud fija de 7 bytes, almacenándose por byte: siglo, año, mes, día, hora, minutos y segundos. También es posible almacenar fechas en forma Juliana, es decir, como un número de días transcurridos a partir de una fecha de referencia. La parte decimal de este número indica la hora de ese día. Las fechas que puede representar tienen un rango desde el primero de enero de 4712 A.C. hasta el 31 de diciembre de 4712 D.C.

RAW. Es similar al tipo char, pero ORACLE no interpreta su significado. Podría servir para almacenar información binaria como la utilizada para representar gráficos. Es posible tener también un tipo LONG RAW, el cual es para el tipo LONG, lo que RAW es para el tipo CHAR.

Soporta tipos definidos en ambientes IBM para tener compatibilidad con estos productos, pero estos tipos son almacenados internamente mediante algún equivalente entre los mencionados con anterioridad.

Es posible cargar una tabla de dos maneras diferentes: cuando no se tiene un archivo de datos previo, se crea la tabla y dinámicamente se va llenando mediante instrucciones de inserción.

Si ya se tiene un archivo de datos y éstos se desean cargar a la base, primero debe crearse la tabla y luego procesarse la utilería ODL (ORACLE Data Loader). Esta utilería recibe como entrada dos archivos:

- un archivo de datos,
- un archivo de control en el cual se definen: una estructura del registro en los datos de entrada; las características del archivo de entrada como dispositivo (cinta o disco); nombre y longitud de registro; la forma en que va a asociar las columnas en la tabla con cada campo del archivo de entrada o con valores constantes.

Para hacer estas definiciones, en el archivo de control deben proporcionarse comandos con una sintaxis específica pero sencilla (DEFINE RECORD... , DEFINE SOURCE... , FOR EACH RECORD INSERT INTO...).

Además permite algunos calificadores que sirven para realizar alguna acción en particular, como sólo cargar determinado número de registros, saltar los "n" primeros registros, detenerse al ocurrir determinado número de errores o hacer efectivas las inserciones cada "n" registros procesados.

Como salidas de esta utilería, se tiene la tabla cargada en ORACLE, un archivo de bitácora en el cual se informan estadísticas del proceso y un archivo de errores en el cual aparecen todos los registros que no pudieron ser cargados.

3.2.1.4 Métodos de acceso y recuperación

ADABAS

Cuenta principalmente con dos tipos de acceso que son secuencial y aleatorio.

El método secuencial se encuentra dividido en tres tipos: lectura física, lectura por ISN y lectura lógica.

Lectura física. Se efectúa un acceso secuencial sobre el DATA STORAGE. Los datos son leídos exactamente en el orden en que están almacenados en la base de datos.

Lectura por ISN. Los registros son leídos en orden del ISN (Internal Sequential Number). En este método se efectúa un acceso al convertidor de direcciones y al DATA STORAGE. El convertidor de direcciones entrega el número de bloque donde se encuentra almacenado el registro en el DATA STORAGE.

Lectura lógica por descriptor. Los registros son leídos en el orden de los valores de un descriptor. En este método se efectúa un acceso hacia las listas invertidas, al convertidor de direcciones y al DATA STORAGE. Las listas invertidas entregan todos los ISN de los registros que contienen ese descriptor.

Los métodos anteriores se efectúan con la instrucción READ de NATURAL².

En el método aleatorio se realiza una búsqueda de los registros requeridos a través de uno o varios descriptores y

² NATURAL es el lenguaje de Manipulación y Definición de datos empleado por ADABAS.

retorna el conjunto de ISN que contienen al descriptor. Se efectúa un acceso hacia las listas invertidas, el área de WORK y el DATA STORAGE. Este método se ejecuta con la instrucción FIND.

También existe otro tipo de acceso que es llamado HISTOGRAM. Este método consiste en leer todos los valores de la base de datos que fueron definidos como descriptor, subdescriptor o superdescriptor. Estos son leídos directamente de las listas invertidas y se retorna tanto su valor en orden ascendente como la cantidad de valores de ISN que contienen a ese descriptor.

ORACLE

La forma natural que emplea ORACLE para tener acceso a los datos es secuencial, haciendo un recorrido a través de toda la tabla. Sin embargo existen técnicas como son los índices y los "clusters" que le ayudan a hacer los accesos más eficientes.

El principal objetivo de los índices es incrementar el desempeño y tener un medio con el cual se pueda garantizar el valor único de los campos.

Los índices representan una llave por medio de la cual es sustancialmente más rápido localizar un registro de la base de datos. Se puede elegir como índice a una columna o un conjunto de columnas con las que deseamos localizar un rengion específico en la tabla. Un índice debe de formar parte del predicado de la cláusula WHERE en un SELECT para que la búsqueda sea efectuada a través de éste.

Los índices forman una estructura arborescente del tipo B+. Esta estructura contiene los índices y la posición física del registro "rowid".

La creación de un índice se realiza con la instrucción CREATE INDEX de SQL, indicándole la tabla sobre la cual se creará, la columna o columnas que lo constituirán y si estará o no comprimido. Un índice puede contener hasta 16 columnas, sin embargo la mayor limitante depende del número de caracteres ya que éste debe ser menor de 240. El tamaño de un índice es igual a la suma del ancho de todas las columnas que lo constituyen más un byte separador entre cada columna.

Otra de las ventajas de usar los índices, es que éstos son lógicamente y físicamente independientes de los datos y pueden ser creados y borrados en cualquier momento sin afectar la aplicación. Además, se mantienen automáticamente y el tiempo de recuperación de los datos no aumenta conforme la tabla crece.

El "cluster" es un agrupamiento de tablas físicamente contiguas. Los "cluster" son usados para disminuir el tiempo de respuesta cuando se consultan dos o más tablas que tienen un campo en común, es decir, cuando se hacen "joins".

Otra ventaja de estos agrupamientos es que disminuyen el espacio requerido en disco, ya que las columnas comunes de cada tabla son almacenadas sólo una vez.

Una vez que se ha creado el "cluster", el mantenimiento y uso es totalmente transparente para el usuario y todas sus aplicaciones. Además, es necesario mencionar que cuando es

generado se crea automáticamente un índice formado por los campos comunes de las tablas.

Existen algunas restricciones para la creación de "clusters". Se pueden agrupar como máximo 32 tablas o 10 columnas y al menos una de las columnas que lo formen no debe permitir valores nulos.

3.2.1.5 Control de acceso múltiple

ADABAS

Las técnicas con las que cuenta ADABAS asegura que los datos no pierdan integridad en un ambiente de acceso múltiple, en el cual puede haber actualizaciones simultáneas.

Para hacer esto, los registros que se están actualizando deben estar previamente apartados. Ningún usuario puede modificar un registro apartado, pero sí consultarlo.

Existen cuatro tipos de usuario, los cuales se describen a continuación.

ACCESS ONLY (solo acceso). Este tipo de usuario sólo puede consultar la información sin modificarla.

UTILITIES (utilerías). Es el usuario que maneja la información a través de una utilería.

EXCLUSIVE UPDATE USERS (Usuarios de actualizaciones exclusivas). Son usuarios que al actualizar un archivo impiden que otros realicen actualizaciones en éste.

TRANSACTIONS USERS (Usuarios de transacciones). Son la mayoría de los usuarios. Estos no pueden trabajar en forma exclusiva en el archivo, deben apartar los registros que están actualizando y posteriormente liberarlos. Sus actualizaciones se hacen en forma de unidades llamadas transacciones. Una transacción es un conjunto de movimientos en la base de datos que no pueden quedar interrumpidos. Si antes de su culminación sucede alguna falla, un proceso de respaldo automático es invocado y se eliminan todos los movimientos hechos desde la última transacción completa hasta el momento de la falla. Solamente al procesarse el comando ET (End Transaction), se llevan a cabo los cambios y se liberan los registros.

El usuario puede invocar el proceso de respaldo explícitamente deshaciendo los movimientos efectuados desde el último ET.

El hecho de apartar registros para uso propio, trae como consecuencia la posibilidad de un "deadlock". Para resolver esto, ADABAS cuenta con el concepto de "Transaction Time Limit" (TT), el cual es el tiempo máximo en que puede ser realizada una transacción. Cuando ADABAS detecta que un proceso ha sobrepasado su TT, invoca al proceso de autorespaldo y libera los registros que tenía apartados.

ORACLE

Un elemento muy importante para controlar el acceso múltiple es el archivo denominado BEFORE IMAGE FILE. Este archivo contiene las imágenes anteriores de los registros que serán modificados. Es usado para permitir que todos los usuarios vean la misma información, pues cuando es consultado un registro que está

siendo actualizado, lo que realmente se está viendo es su imagen de antes, almacenada en este archivo.

ORACLE controla el acceso múltiple apartando ya sea un registro o una tabla cada vez que se desea modificar la base de datos. La forma de apartar estos registros es colocando candados. Un candado se inicia explícita o implícitamente a través de un comando y termina con el comando COMMIT o ROLLBACK que hace permanentes los cambios o los deshace, respectivamente. Existen tres diferentes tipos de candados, los cuales se explican a continuación.

SHARE. Sirve para hacer consultas consistentes. Es decir, garantiza que los datos no cambiarán entre dos consultas. Previene que otros usuarios modifiquen la información que se está manejando. Se maneja a nivel de tabla y se obtiene explícitamente mediante el comando LOCK TABLE.

EXCLUSIVE. Sirve para modificar la información, imposibilitando a otros usuarios a que lo hagan. Aunque otros usuarios pueden consultar la información, no se garantiza que sea la actual. Se maneja a nivel de tabla. Se obtiene implícitamente usando los comandos INSERT, UPDATE o DELETE o explícitamente usando el comando LOCK TABLE.

SHARE UPDATE. Es el único de los tres que se maneja a nivel renglón. Sirve para tener el derecho de hacer modificaciones a un renglón o conjunto de ellos. No aparta toda la tabla, pero nadie puede modificar los renglones apartados. Se obtiene implícitamente a través del comando SELECT...FOR UPDATE, o explícitamente mediante el comando LOCK TABLE. Su uso es muy conveniente pues aumenta la concurrencia y garantiza que los

registros no cambien desde su consulta hasta que son actualizados. En el momento de hacer la actualización, la tabla es puesta en modo EXCLUSIVE, por lo tanto es conveniente hacer las actualizaciones justo antes del COMMIT. De hecho, algunos productos de ORACLE, como SQL*FORMS utilizan este tipo de candados para aumentar la concurrencia.

ORACLE detecta "deadlocks" y los resuelve, ejecutando un ROLLBACK a la transacción que tiene el menor número de bloques modificados.

3.2.1.6 Límites del SMOB

ADABAS

Tiene los siguientes límites:

# archivos	255
# registros/archivo	16.7 millones
# tamaño del registro	10,000 bytes (no comprimido) 3000 bytes (comprimido)
# campos/registro	500
# campos múltiples	191
# grupos periódicos	99
# descriptores/archivo	200

ORACLE

Sus límites se muestran a continuación:

# tablas	9,500
# tablas/cluster	32
# columnas/tabla	254
# caracteres/columna	240 (tipo CHAR) 65,535 (tipo LONG)

■ registros/tabla	2 a la 29 bloques
■ bloques/tabla	2 a la 29 bloques
■ tamaño de la b.d.	2 a la 29 bloques
■ índices columnas/tabla	0 hasta n (n columnas)
■ bytes renglón	128 KB
■ caracteres/SQL Statment	8,000 bytes

Los bloques indicados anteriormente, son bloques ORACLE.

3.2.2 Características y facilidades para el usuario

3.2.2.1 Facilidades de consulta

ADABAS

NATURAL es el lenguaje empleado por ADABAS para efectuar consultas a la base de datos. Las consultas se efectúan a través de las instrucciones READ, FIND y GET. Con éstas se pueden realizar las siguientes operaciones:

- Efectuar lecturas secuenciales.
- Realizar búsquedas con criterios específicos.
- Encontrar determinados registros de acuerdo a una llave o descriptor.
- Efectuar diferentes acciones al encontrar el fin de un archivo.
- Efectuar diferentes acciones al no encontrar determinados registros.
- Encontrar el número de registros que cumplen con una condición.
- Encontrar los ISM (Internal Sequential Number) que cumplen con una condición.

Además, cuenta con un conjunto de funciones que auxilian en la manipulación de los datos. En general se dividen en las siguientes: grupales como SUM, OLD, TOTAL, MAX, MIN, AVRC, etc.; matemáticas como SQR, ABS, ATN, SIN etc.; para procesamiento de ciclos como COUNT y NCOUNT; para manejar las fechas en diferentes formatos y para consultar tiempos de proceso.

ORACLE

SQL es el lenguaje que utiliza ORACLE para tener acceso a la base de datos. Las consultas se efectúan a través del comando SELECT con el cual se indica:

- las columnas que queremos desplegar,
- la tabla o tablas de donde se desea consultar la información,
- los criterios para seleccionar determinados renglones,
- el orden en el que se entregarán los renglones de la tabla,
- si se desea hacer agrupamiento de renglones en base a un campo.

Además, SQL tiene una gran variedad de funciones que se pueden aplicar al momento de la consulta: aritméticas como ABS, TRUNC, ROUND etc.; grupales, que actúan sobre un conjunto de valores, como AVG, COUNT, MAX, MIN, SUM; funciones que manipulan los caracteres, como LENGTH, SUBSTR, INITCAP etc. y funciones que manipulan la fecha como ADD_MONTHS, LEAST, LAST_DAY, TO_CHAR etcétera.

Mediante SQL es posible efectuar todas las operaciones tanto del álgebra como del cálculo relacional, como son intersecciones, proyecciones, selecciones, "joins" y uniones.

3.2.2.2 Facilidades de actualización

ADABAS

NATURAL contiene comandos que permiten insertar, actualizar y borrar información de la base de datos.

Mediante el comando STORE se mueve el contenido de memoria temporal hacia la base de datos. Los comandos UPDATE y DELETE permiten modificar y cambiar el contenido de la base de datos.

Para llevar a cabo realmente las modificaciones en la base de datos se debe de ejecutar el comando ET.

ORACLE

Los comandos de SQL permiten insertar uno o mas renglones hacia una tabla (INSERT), actualizar o modificar datos (UPDATE) y borrar los renglones de la tabla (DELETE).

Estas instrucciones forman parte del LMD (Lenguaje de Manipulación de Datos), son muy poderosas y tienen una sintaxis muy sencilla. Permiten tener consultas anidadas para seleccionar un conjunto de renglones sobre los cuales pueda operar.

3.2.2.3 Diccionario de datos

ADABAS

Cuenta con un diccionario de datos activo en línea llamado PREDICT. PREDICT está totalmente integrado con la base de datos, así como con NATURAL.

Es un producto de gran importancia en este ambiente ya que es un enlace entre las estructuras físicas de los archivos y las representaciones lógicas de los datos.

Funciona en base a menús, con una gran cantidad de opciones que permiten entre otras cosas:

- generar vistas lógicas,
- dar mantenimiento a la base de datos, archivos, campos, llaves, programas etc.,
generar referencias sobre los programas que usan determinados campos,
- controlar la seguridad de la información y definir estándares, es decir, campos cuyo cambio en formato repercute en todos los archivos que los manejan,
- obligar a realizar la documentación para datos de entrada de acuerdo a las necesidades.

Una de sus principales ventajas es la total integración que tiene con NATURAL, lo que permite que programas escritos en este lenguaje tengan acceso a la información del diccionario de datos, y que todos los recursos y datos que ocupan queden registrados en forma automática.

ORACLE

El diccionario de datos es un conjunto de tablas y vistas que contienen información de la base de datos y de los objetos que existen en ésta, tales como tablas, columnas, vistas, índices, "clusters", usuarios y privilegios de acceso.

Estas tablas y vistas se crean en el momento de la instalación y se actualizan automáticamente. El ABD puede generar

nuevas vistas a partir de las tablas existentes, pero lo normal es no hacerlo. Un usuario no puede alterar el contenido de estas tablas.

Dado que el diccionario de datos es un conjunto de tablas, la forma de tener acceso a él es mediante comandos de SQL. Esto da mucha flexibilidad pues en cualquier momento la información puede consultarse haciendo selecciones y proyecciones de estas tablas.

Mediante el comando COMMENT es posible hacer comentarios de las tablas y columnas. Estos comentarios quedan almacenados en el diccionario y pueden ser consultados posteriormente.

Cuarenta y cuatro tablas y vistas componen este diccionario. A continuación se mencionan las más importantes.

SYSCATALOG. Contiene descripciones de todas las tablas, vistas y "clusters" que pertenecen al usuario o a las que éste tiene acceso.

CATALOG. Contiene descripciones de todas las tablas, vistas y "clusters" que pertenecen al usuario o a las que éste tiene acceso, pero sin incluir las del diccionario de datos.

TAB. Contiene las descripciones de todas las tablas, vistas y clusters que pertenecen al usuario.

SYSCOLUMNS. Contiene los nombres y descripciones de columnas de tablas, vistas y "clusters" que pertenecen al usuario o a las que éste tiene acceso.

COLUMNS. Contiene los nombres y descripciones de columnas, ya sea de tablas, vistas y "clusters" que pertenecen al usuario o a las que éste tiene acceso, pero sin incluir las del diccionario de datos.

COL. Contiene los nombres y definiciones de columnas de tablas, vistas y clusters que pertenecen al usuario.

SYSVIEWS. Contiene los comandos que definieron las vistas que tiene el usuario o que puede tener acceso.

IEWS. Contiene los comandos que definieron las vistas que tiene el usuario.

INDEXES. Contiene información de los índices que el usuario ha creado y de aquellos que han creado otras personas sobre tablas de ese usuario.

STORAGE. Contiene información del espacio ocupado por las tablas e índices que tiene el usuario.

SYSTABAUTH. Almacena información acerca de los privilegios de acceso sobre tablas, vistas y clusters que el usuario ha otorgado o que le han sido otorgados.

SYSUSERLIST. Contiene a los usuarios de ORACLE.

DTAB. Describe las tablas y vistas que forman el diccionario de datos.

3.2.2.4 Interacción con otros lenguajes

ADABAS

Interactúa con lenguajes de tercera generación a través de una interfaz de alto nivel que es ADASQL.

Los comandos de ADA SQL son usados dentro del programa fuente del lenguaje de tercera generación: Únicamente hay que anteponer a cada comando un mnemónico necesario para el precompilador. Este programa es procesado a través del precompilador de ADASQL el cual traduce los comandos de SQL a llamadas directas. Posteriormente, el programa es compilado, ligado y ejecutado en su forma estándar.

ADASQL esta constituido por comandos que ayudan a manipular, definir, consultar y proteger los datos.

Solo las funciones de manipulación de datos de SQL están incorporadas a ADASQL, ya que las funciones de definición y seguridad de los datos son ejecutadas por PREDICT y las funciones de consulta son ejecutadas por SUPERNATURAL.

ORACLE

Cuenta con un conjunto de precompiladores que permiten manipular la base de datos a través de otros lenguajes. Todos ellos se encuentran clasificados dentro de los productos PRO de ORACLE.

Estos precompiladores forman una interfaz entre la base de datos y los lenguajes de tercera generación. De esta forma se pueden desarrollar aplicaciones en un lenguaje de alto nivel usando los datos de ORACLE. Los precompiladores con los que

cuenta son: PRO*FORTRAN, PRO*COBOL, PRO*ADA, PRO*PLI, PRO*C y PRO*PASCAL.

Así, un programa escrito en lenguaje de tercera generación que tiene acceso los datos de ORACLE, puede usar cualquier instrucción de SQL, únicamente es necesario anteponerle un mnemónico del precompilador.

También es posible efectuar llamadas de bajo nivel usando las rutinas de interface OCI (ORACLE Call Interface). Esta es la forma más rudimentaria de tener acceso a los datos a través de un lenguaje de tercera generación.

3.2.2.5 Productos adicionales

ADABAS

Los productos que conforman el ambiente de la base de datos en ADABAS son : ADABAS, NATURAL, PREDICT y otros productos específicos.

NATURAL. Es un producto totalmente compatible con ADABAS y PREDICT. Este es un lenguaje de cuarta generación que permite manejar la información de la base de datos. A través de él se pueden realizar las siguientes operaciones:

- desarrollar aplicaciones,
- generar reportes y consultas interactivas,
- permitir o denegar el acceso a NATURAL,
- generar formas de entrada.
- hacer llamadas desde lenguajes de 3a. generación,
- emplear funciones de librerías.

PREDICT. Es un diccionario de datos interactivo, una descripción más detallada de este producto aparece en el tema referente al diccionario de datos.

NATURAL2. Es la segunda versión de NATURAL y proporciona notables mejoras sobre la versión anterior, como son: manejador interactivo de pantallas, lenguaje de programación interactivo, generador de reportes etcétera. Dentro de NATURAL2 se encuentran:

- **NATURAL/ELITE.** Permite aprender Natural en forma interactiva.
- **SuperNATURAL.** Permite dar acceso a los datos en una forma más fácil y con mayor flexibilidad.
- **NATURAL GRAPHICS.** Toma información de la base de datos y la traslada a gráficas.
- **NATURAL SECURITY.** Provee un sistema de seguridad para usuarios, base de datos, archivos y campos.
- **ADASQL.** Es una interface de lenguajes de tercera generación con ADABAS.

ORACLE

ORACLE cuenta con un conjunto de productos que facilitan el uso de la base de datos. Estos productos se encuentran divididos en tres tipos de acuerdo a su función y son: productos EASY, productos SQL y productos PRO.

EASY. Es un producto basado en menues, los cuales guían, auxilian e informan en línea. Este producto está enfocado a usuarios casuales o usuarios que no están familiarizados con computadoras.

SQL. Indica los productos interactivos que están enfocados a usuarios que tienen cierto conocimiento de bases de datos; éstos conforman los principales productos de ORACLE. A continuación se presenta una breve descripción de cada uno de ellos:

- SQL*PLUS. Con este producto se puede consultar y modificar la base de datos así como obtener reportes de ella.
- SQL*FORMS. Es un manejador de formas de pantalla, que permite crear, modificar y usar pantallas tanto de entrada como de salida.
- SQL*CALC. Es una hoja de cálculo completamente integrada con los datos de ORACLE.
- SQL*MENU. Crea una interfaz entre diferentes programas y comandos del sistema operativo para manejarlos bajo un mismo ambiente en base a menús.
- SQL*GRAPH. Permite graficar la información de la base de datos.
- SQL*REPORT. Permite generar reportes complejos de la información de la base de datos.
- SQL*NET. Permite tener acceso a la base de datos en un ambiente de red.

PRO. Es un conjunto de productos que proporcionan una interfaz con el SMD. Estos productos requieren usuarios expertos en programación y con gran experiencia tanto en SQL como en los productos de ORACLE. Son los siguientes:

- Pro*C.
- Pro*COBOL.
- Pro*FORTRAN.

- PROFPL/1.
- PROFPSICAL.
- PROFADA.

3.2.3 Características de administración

3.2.3.1 Seguridad

ADABAS

Existe una utilería llamada NATURAL SECURITY, la cual permite controlar el acceso a la información desde NATURAL. NATURAL SECURITY trabaja en tres niveles: nivel usuario, nivel aplicación y nivel vistas.

Nivel usuario. Restringe el acceso a usuarios creando archivos que incluyen: la identificación de un usuario y su contraseña, aplicaciones disponibles para él, así como las utilerías y comandos que puede usar.

Nivel aplicación. A este nivel se puede definir: el periodo de tiempo en el cual puede ser ejecutada una aplicación, si la aplicación es pública o privada y si la aplicación está restringida a que sólo la use un usuario, una terminal o ambos.

Nivel vistas. Se puede definir (en conjunción con PREDICT), si una vista es pública o privada. Las privadas se pueden restringir a un usuario en particular o a un grupo de usuarios.

Existe otro tipo de control de seguridad. Este es directamente a través de ADABAS. Tiene dos niveles: nivel valor y nivel archivo.

Nivel Valor. Un usuario puede definir valores posibles o rango de valores para algún campo, de modo que no se permita la modificación del campo cuando estos valores no son válidos.

Nivel Archivo. Mediante la utilidad ADASCR es posible controlar el acceso a determinados archivos. Esto se hace asociándoles un nivel de protección de 0 a 15 (donde 0 es la más baja). También debe definirse contraseñas asociadas a distintos archivos con un nivel de autorización de 0 a 14. Estos niveles se definen en forma separada para consulta o actualización. Para que un usuario pueda tener acceso algún archivo protegido, debe conocer una contraseña asociada a éste con un nivel de autorización mayor o igual que el del archivo, ya sea para consultarlo o actualizarlo. Un archivo con nivel 0 es público y con nivel 15 es inaccesible para todos.

ORACLE

Para entrar a ORACLE es necesario tener una cuenta y una contraseña. Estos son independientes de la cuenta del sistema operativo.

Existen tres tipos de usuarios, cada uno con diferentes privilegios. A continuación se mencionan los privilegios por orden de categoría de menor a mayor, indicando las operaciones que permiten realizar.

CONNECT

- # Puede conectarse a ORACLE.
- # Tiene permitido ver datos de otros usuarios, si se tiene la autorización de SELECT.
- # Puede modificar otros datos (INSERT, DELETE o UPDATE) si éste tiene las autorizaciones respectivas sobre ellos.

- 1 Puede crear vistas y sinónimos.
- 2 No puede crear tablas, "clusters", ni índices.

RESOURCE

- 1 Tiene todos los privilegios del usuario CONNECT.
- 2 Tiene permitido crear tablas, índices y "clusters".
- 3 Puede dar o quitar autorizaciones sobre estos objetos.
- 4 Puede usar el comando AUDIT para controlar el acceso a estos objetos.

DBA

- 1 Tiene todos los privilegios del usuario RESOURCE.
- 2 Tiene acceso a todos los datos y puede dar cualquier instrucción de SQL sobre ellos.
- 3 Puede dar y quitar privilegios a otros usuarios.
- 4 Puede crear sinónimos públicos.
- 5 Puede crear y alterar particiones.
- 6 Tiene permitido controlar las opciones predeterminadas para auditar.
- 7 Puede exportar la base de datos completa.

ORACLE posee una forma de monitorear la seguridad de la base de datos. El DBD puede habilitar la auditoria del sistema. De esta forma se almacenan en tablas del diccionario de datos las diferentes opciones que se quieren rastrear. El comando AUDIT de SQL tiene dos opciones: una de ellas permite al DBD vigilar objetos de la base en general, como acceso a la base de datos, creaciones de tablas, borrado de tablas, uso del comando GRANT o REVOKE para alterar los privilegios, etcétera. La segunda forma permite a un usuario en general rastrear el acceso (mediante consultas, actualizaciones, inserciones) a alguna tabla o vista de su propiedad. La información producida

se almacena en tablas y puede consultarse con el comando SELECT.

El propietario de una tabla es el único capaz de tener acceso a ella (excepto un ABD). Sin embargo, un usuario puede otorgar autorización para que se tenga acceso a su tabla mediante SELECT, INSERT, UPDATE, DELETE, ALTER o INDEX. Este acceso puede ser otorgado a algún usuario en particular o a todos con la opción PUBLIC. Así mismo, un usuario puede crear vistas de su tabla, dejando ver sólo las columnas o renglones que él desee.

En general, para entrar a cualquier producto ORACLE, es necesario teclear un USERNAME y un PASSWORD. Sin embargo, si en un momento dado resulta demasiado tedioso, es posible que el ABD asigne a un usuario un acceso automático, de modo que éste no necesite identificarse cada vez que entre a SQLPLUS. De igual forma, es posible que un usuario que entró a ORACLE con un USERNAME, se conecte a otra clave (conociendo su USERNAME y PASSWORD) sin necesidad de salir a nivel sistema operativo.

3.2.3.2 Respaldo y recuperación de errores

ADABAS

Existen en ADABAS diferentes utilerías que permiten conservar la consistencia de los datos. A continuación se mencionan algunas de ellas.

ADABCK. Mediante esta utilería se pueden efectuar respaldos y restauraciones de la base de datos. Cuenta con la función DUMP que permite respaldar total o parcialmente la base de datos y

con la opción RESTORE para restablecerla. Además permite con diferentes funciones desplegar información de los respaldos que se efectuaron con el DUMP, indicar el número de copias físicas que se generarán de la información respaldada, crear un nuevo archivo de respaldo PLOG (Protection Log File) o trabajar con el archivo anterior.

ADAVFY. Con esta utilidad es posible verificar la consistencia de la base de datos. Cuenta con diferentes funciones que permiten elegir entre la verificación del convertidor de direcciones o verificación de un número de archivos específicos.

ADAREC. Con esta utilidad, es posible recuperar la información de la base de datos en caso de que se presente alguna falla en los dispositivos que la contienen o en caso de que una aplicación destruya información importante. Cuenta principalmente con dos funciones COPY y REGENERATE. La función COPY permite determinar mediante el archivo PLOG los bloques de datos válidos y con REGENERATE es posible recuperar la información de la base total o parcialmente.

ORACLE

Para respaldar la información de la base de datos ORACLE cuenta con las utilidades EXPORT e IMPORT. Estas utilidades permiten realizar las siguientes operaciones:

- Almacenar los datos de ORACLE en un archivo del sistema operativo.
- Almacenar las definiciones de tablas, accesos, sinónimos, definiciones de vistas y definiciones de espacios.
- Transportar datos entre diferentes bases de datos ORACLE.

- Transportar los datos de una versión a otra.
- Mover datos de un propietario a otro.
- Reconstruir la base de datos a partir de un archivo previamente exportado.

Estas utilerías son fáciles de ejecutar y en forma interactiva preguntan por los diferentes requerimientos del usuario.

Para asegurar la integridad de la información, ORACLE maneja el concepto de transacciones o unidades lógicas de trabajo. Estas transacciones funcionan como una simple entidad que no puede ser dividida. Esto quiere decir, que si ocurre alguna falla cuando no se ha terminado la transacción, esta debe de fallar totalmente y para ser exitosa, todas y cada una de sus operaciones tuvieron que haber terminado satisfactoriamente. Esto asegura la consistencia de la información en caso de que el sistema operativo o una aplicación de usuario fallen.

Una transacción en ORACLE puede estar formada desde una simple operación de SQL hasta todo un programa que realice un conjunto de operaciones en la base de datos.

Todas las operaciones de un usuario están divididas en transacciones o unidades lógicas de trabajo. Estas comienzan cuando la primera instrucción ejecutable de SQL es encontrada y terminan cuando se ejecuta un COMMIT, un ROLLBACK, cuando se usa alguna instrucción DDL (CREATE, DROP, RENAME y ALTER), cuando ocurren "deadlocks" o cuando sucede alguna terminación anormal.

Cuando una unidad de trabajo termina, inmediatamente comienza la siguiente unidad de trabajo. Si nunca se encuentra

un COMMIT o ROLLBACK la unidad de trabajo se termina al finalizar el programa.

La instrucción COMMIT hace definitivas las operaciones efectuadas en una unidad lógica de trabajo. El ROLLBACK deshace todas las operaciones que se estaban realizando en la unidad lógica de trabajo. Es necesario mencionar que una vez que se efectúa un COMMIT, esta transacción o unidad lógica de trabajo ya no puede ser recuperada por un ROLLBACK.

La utilidad del BEFORE IMAGE FILE (BI) es permitir, mediante el comando ROLLBACK deshacer los cambios efectuados a la base de datos antes de un COMMIT. Al hacer esto, los registros almacenados en el BI son escritos nuevamente en la base de datos.

El BI es un archivo de tipo circular, es decir que este archivo puede ser reutilizado escribiendo sobre aquellos bloques que ya no están siendo usados. Puede ser monitoreado a través de la utilidad ODS de ORACLE.

ORACLE cuenta también con una utilidad llamada AIJ (After Image Journal) que permite recuperar la base de datos en caso de que suceda alguna falla a los dispositivos que la contienen. Con ésta es posible reproducir una copia exacta de la información perdida a través de un conjunto de archivos de bitácora donde se registran los cambios hechos hasta el momento.

Si el proceso de bitácora está habilitado, todas las transacciones son escritas secuencialmente a un conjunto de archivos antes de que estas sean escritas a la base de datos. Si llegara a ocurrir alguna falla, es posible invocar a esta

utilería para recuperar la información a partir de un respaldo anterior y de los archivos de bitácora generados.

3.2.3.3 Facilidades para monitorear

ADABAS

Mediante el comando DISPLAY de la utilería ADADPR es posible monitorear lo que está ocurriendo en la base de datos. Este comando despliega la siguiente información:

- Los comandos usados.
- Los recursos apartados.
- Los parámetros actuales.
- Los usuarios.
- El estado de la base de datos.

Al desplegar el estado de la base de datos, esta utilería proporciona la siguiente información:

- Información general, como número de la base de datos, número de procesador y versión.
- Número de interrupciones actuales y totales.
- Procesos en estado de espera.
- "Deadlocks".
- Parámetros de eficiencia.
- Número de entradas y salidas tanto para lectura como para escritura.
- Frecuencia de cada comando ADABAS.

Existen otras utilerías que aunque no monitorean, dan información respecto al espacio en disco. Estas son: ADADEV y ADAREP.

ADADEV indica el número de bloques malos, así como cuantos y cuales cilindros tiene apartados.

ADAREP genera información sobre:

- Cantidad y localización del espacio apartado y disponible para el ASSOCIATOR y DATA STORAGE.
- Resumen sobre los archivos existentes en la base de datos.
- Puntos de verificación.
- Cada archivo, como espacio apartado, espacio disponible, número de registros cargados, máximo ISN y definición de campos, entre otros datos.

ORACLE

Posee una utilería llamada DDS (ORACLE Display System), que permite monitorear la actividad que está teniendo la base de datos y contabilizar todo lo que ha pasado desde la última vez que se levantó.

Básicamente informa acerca de:

- Qué procesos están usando ORACLE.
- Qué programas de ORACLE se están usando (SQL*PLUS, SQL*FORMS, etcétera.).
- A qué tablas se está teniendo acceso.
- Qué cantidades se están ocupando o esperando.
- El estado actual del BI.
- La actividad de entrada/salida tanto lógica como física.

Es una utilidad muy completa que proporciona bastante información para poder en un momento dado sintonizar el sistema o usar una política que mejore el desempeño. Maneja diferentes pantallas y la información puede mandarse a un archivo de salida.

3.2.3.4 Utilerías para administración en general

ADABAS

Tanto ADABAS como NATURAL cuentan con utilerías que permiten controlar los recursos y operar la base de datos cotidianamente. Entre las principales podemos mencionar a ADAOPR y ADADBM, además de las utilerías de monitoreo y reportes que ya se han mencionado.

ADAOPR tiene opciones para operar el núcleo de la base de datos, como asignar los tiempos límite de transacción a un valor específico, poner candados a un archivo, detener la base de datos, monitorear, etcétera.

ADADBM permite hacer modificaciones a la base de datos, como renombrar, reenumerar o borrar archivos, crear nuevos archivos para el DATA STORAGE y ASSOCIATOR y ver o eliminar procesos que hayan quedado pendientes.

A continuación se menciona la forma de efectuar las operaciones de alta a usuarios, revania, y dar de baja la base de datos:

Procedimiento para dar de alta la base de datos. Sólo es necesario correr algunos procedimientos de comandos que

instalan en memoria el núcleo de ADABAS, hacen las definiciones requeridas y arrancan un proceso (mediante la utilería ADANUC) que estará funcionando todo el tiempo que la base de datos esté activa. El procedimiento de comandos que activa la base de datos es llamado SYADAINI.COM.

Procedimiento para dar de baja la base de datos. Cuenta con la utilería ADAOPR con la opción SHUTDOWN, la cual termina el proceso arrancado durante la fase de levantamiento. Además es conveniente efectuar un proceso que desinstala las imágenes cargadas en memoria principal.

Procedimiento para dar de alta a un usuario. Para que un usuario pueda tener acceso a la base de datos a través de NATURAL es necesario que éste haya sido dado de alta previamente. Esto se hace a través de la utilería de NATURAL NATPARM, la cual crea un archivo con todas las características que tendrá la cuenta. Cada vez que se entra a NATURAL, el sistema busca un archivo en el directorio SYS\$NATPARM con el nombre del USERNAME asignado en VMS, si lo encuentra lee las características asignadas y permite el acceso, si no lo encuentra el acceso es denegado.

ORACLE

ORACLE tiene utilerías que facilitan la administración y operación diaria de la base de datos. A continuación se mencionan:

Procedimiento para dar de alta la base de datos. Se emplea la utilería IOR. Existen dos formas de darla de alta, éstas son las siguientes:

- ¶ IOR INIT. La opción INIT inicializa la base de datos sin realizar un ROLLBACK y sin verificar que usuarios se encuentran en ella. Esta opción es utilizada la primera vez que se instala ORACLE y cuando se actualiza su versión. Previamente debe hacerse un respaldo, pues todos los datos son eliminados.
- ¶ IOR WARM. La opción WARM es la forma normal de levantar la base de datos. Esta opción abre los archivos y realiza un ROLLBACK de la última transacción que no fue completada.

Esta utilidad despliega información del tamaño de los componentes del SGA como son: números de bytes de los "buffers" de la base de datos, número de bytes de datos variables, número de "buffers" del BI, número de "buffers" de la base de datos, etcétera.

Procedimiento para dar de baja la base de datos. Para dar de baja la base de datos existen dos formas y son:

- ¶ IOR SHUT. La opción SHUT no permite entrar a la base de datos a los nuevos usuarios, pero no afecta a los usuarios actuales. En esta opción el sistema espera a que los usuarios salgan de ORACLE y posteriormente se da de baja la base de datos. Cuando se inicia este proceso todas las transacciones pendientes son completadas antes de terminar. Esto quiere decir que al iniciar la base de datos en el caso de un IOR WARM no requerirá de un ROLLBACK para las transacciones pendientes.
- ¶ IOR CLEAR. La opción CLEAR detiene todos los procesos sin esperar a que los usuarios salgan del sistema y no efectúa las transacciones pendientes. De esta forma el siguiente

WARN que se ejecute activará el ROLLBACK terminando con todas las transacciones que hayan quedado incompletas.

Procedimiento para dar de alta a un usuario. Para usar la base de datos es necesario tener asignada una cuenta con una contraseña. Para dar de alta una cuenta en ORACLE se usa el comando GRANT de SQL. En éste se le indica el tipo de usuario (CONNECT, RESOURCE o DBA), el nombre de la cuenta (USERNAME) y la contraseña (PASSWORD).

3.2.4 Facilidades que ofrece el proveedor

3.2.4.1 Soporte

ADABAS

El Centro de Cálculo no tiene actualmente ningún tipo de contrato referente a soporte técnico con la Compañía que distribuye el producto.

ORACLE

La Compañía ORACLE y el Centro de Cálculo no tiene ningún tipo de contrato en lo que respecta a soporte técnico.

3.2.4.2 Calidad de la documentación disponible

ADABAS

La calidad de la información, aunque no es mala, deja un poco que desear en cuanto a su organización y claridad para el usuario. Este es un punto a mejorar en futuras versiones.

ORACLE

Cuenta con diversos manuales para cada uno de sus productos. La presentación es adecuada y la organización es suficientemente didáctica. En general no se encontró problema para consultar la información cada vez que fue necesario.

3.2.5 Requerimientos del SMBD

3.2.5.1 Cantidad de espacio en disco

ADABAS

Los datos cargados en ADABAS requieren espacio tanto en el DATA STORAGE como en el ASSOCIATOR.

En el ASSOCIATOR se requiere espacio en las siguientes cuatro áreas:

- Área de índices normales. Se almacena el más bajo nivel de las listas invertidas.
- Área de índices principales. Se almacenan apuntadores a los índices normales.
- Área de índices superiores. Son apuntadores que pueden ser creados y usados únicamente cuando se requiere.
- Área del Convertidor de direcciones. Se almacena el número de bloque donde está el registro de datos.

Es conveniente mencionar que el tamaño del bloque varía dependiendo del tipo de dispositivo y del área del manejador (DATA STORAGE, ASSOCIATOR o WORK). Por ejemplo: en Discos RPO6 se tienen bloques de 2044 bytes para el ASSOCIATOR y de 5628 para el DATA STORAGE.

El factor de crecimiento para el DATA STORAGE depende en gran medida del factor de sobreflujo del archivo ya que los registros no pueden dividirse en dos bloques. El espacio ocupado en el ASSOCIATOR es debido principalmente a los índices normales y principales. Resultados experimentales serán mostrados posteriormente.

ORACLE

Requiere una cantidad de espacio extra en disco para almacenar cada tabla, renglón y columna.

Cada tabla requiere dos bloques extras. En el primer bloque almacena la dirección de inicio de cada extensión, la dirección del último bloque y cuántos bloques ocupa cada extensión. En el segundo bloque se guarda información relacionada con la definición de la tabla.

Cada renglón requiere de cuatro o cinco bytes extras. Los primeros dos sirven para almacenar la longitud del renglón en bytes, los siguientes dos contienen el número secuencial del renglón y si la tabla está agrupada en "cluster", es necesario un byte adicional.

Cada dato en cada columna necesita dos bytes adicionales. Un byte almacena el número de identificación de la columna y otro el número de bytes de la columna.

ORACLE maneja bloques lógicos de 2048 bytes. De éstos, 1972 son ocupados para datos y 76 para almacenar información interna asociada al bloque físico (en 44 bytes) y al bloque lógico (32 bytes).

A partir de estos datos es posible calcular en forma aproximada el factor de crecimiento de un archivo cuando es cargado en ORACLE. Posteriormente se mostrarán datos experimentales a este respecto.

3.2.5.2 Cantidad de memoria principal

ADABAS

El núcleo de la base de datos corre con un "working set" de 1 Mb. Es conveniente que los usuarios utilicen un "working set" del mismo tamaño.

ORACLE

Toda la información de la base de datos cuando ésta se encuentra activa es almacenada en la SGA (System Global Area). El tamaño de la SGA es establecido por los parámetros del archivo INIT.ORA. Entre los principales aspectos que controlan estos parámetros están el número de "buffers", el número de definiciones de columnas, tablas y "clusters" que puede haber en memoria, así como el número de transacciones activas. Considerando los valores predefinidos, la SGA ocupa alrededor de 400 Kb.

Además de los requerimientos de memoria globales, es conveniente aunque no indispensable que los usuarios tengan un "working set" de 1 Mb. (2048 páginas VMS). Valores menores decrementan el tiempo de respuesta.

3.3 Pruebas experimentales

Para poder comparar cuantitativamente ambos manejadores de bases de datos, se diseñaron pruebas experimentales mediante las cuales se pudiera medir su eficiencia.

Se diseñaron para este fin tres aplicaciones estadísticas sencillas, que implican consultas que cotidianamente se harán en el CECAFI. Estas son las siguientes:

- Obtención del número de alumnos de alguna generación en particular inscritos al semestre actual.
- Obtención del número de alumnos de alguna generación en particular inscritos a cada asignatura.
- Obtención del número de alumnos inscritos a una cantidad específica de asignaturas (0,1...7).

Para el desarrollo de estas aplicaciones se utilizaron dos tablas con información referente a los alumnos de la Facultad de Ingeniería, éstas son: ALUMNO e INSCRITO. A continuación se muestran los programas realizados para consultar estas tablas tanto en DRACLE como en ADABAS, así como los resultados obtenidos.

SMDB: ORACLE

Num. Registros Cargados	Num. E/S	Num. Fallos Pagina	Tiempo Procesador	Tiempo Total
100	107	1245	00:20.76	00:27.22
1000	154	1812	01:44.02	02:02.13
5000	428	2275	08:14.30	09:28.74
10000	2027	2022	16:45.24	19:09.31
19979	2756	3291	34:41.91	40:07.91

SMDB: ADABAS

Num. Registros Cargados	Num. E/S	Num. Fallos Pagina	Tiempo Procesador	Tiempo Total
100	326	1042	00:07.84	00:27.68
1000	376	1350	00:15.07	00:32.54
5000	909	2168	00:36.45	01:14.52
10000	1646	2170	01:03.72	02:09.00
19979	3180	2159	02:01.56	04:00.26

Tabla 3.1 Estadísticas del proceso de carga del archivo alumno.

SMDB: ORACLE

Num. Registros Cargados(3)	Num. E/S	Num. Faltas Pagina	Tiempo Procesador	Tiempo Total
364	214	1453	00:13.95	00:23.81
3642	659	1786	00:52.55	01:28.84
18214	1301	1782	03:40.85	05:41.02
36086	2646	1785	08:55.10	10:25.97
45776	3057	1889	08:58.75	14:19.85

SMDB: ADABAS

Num. Registros Cargados	Num. F/S	Num. Faltas Pagina	Tiempo Procesador	Tiempo Total
100	252	937	00:08.74	00:22.12
1000	313	1088	00:10.94	00:25.65
5000	438	1631	00:20.38	00:38.65
9906	664	1940	00:32.80	00:56.89

Tabla 3.2 Estadísticas del proceso de carga del archivo inscrito.

³ El número de registros cargados para cada manejador es diferente, debido a que su longitud no es la misma. Se está tomando por lo tanto el mismo número de bytes en cada caso.

SMBD: ORACLE

No. Alumnos cargados	Inf. entrada (Kb)	Inf. cargada (kb)	crecimiento (%)
100	24	120	400.00
1000	239	720	201.26
5000	1193	3170	165.72
10000	2379	5520	132.03

SMBD: ADASAS

No. Alumnos cargados	Inf. entrada (Kb)	Inf. cargada (kb)	crecimiento (%)
100	24	496	1975.00
1000	237	466	104.36
5000	1193	1463	23.05
10000	2379	2696	13.27

Tabla 3.3 Estadísticas sobre el espacio ocupado.

```

REM      /***** *****/
REM      /* Programa que obtiene el número de alumnos      */
REM      /* de una generación dada inscritos al semestre  */
REM      /* semestre actual                                */
REM      /* *****/
REM      /* CMBD:                ORACLE                    */
REM      /* *****/
REM      /* Entradas:            generación                */
REM      /* *****/
REM      /* Tablas consultadas: ALUMNO                    */
REM      /*                      INSCRITO                  */
REM      /* *****/
REM      /* Programa:           NUM_ALU.SQL                */
REM      /* *****/
REM      /***** *****/

CLEAR COLUMN
COLUMN sysdate NEW_VALUE fecha
COLUMN sysdate NOPRINT
TTITLE RIGHT fecha -
SKIP CENTER 'NUMERO DE ALUMNOS INSCRITOS EN EL SEMESTRE' &1 -
SKIP 2

SPOOL rep@tesis:sol_num_alu2
SELECT sysdate,count(*) "Numero de Alumnos"
FROM   cl_alumno
WHERE  per_ing_fac = &1
AND    exists
      (SELECT *
      FROM cl_inscrito
      WHERE cl_alumno.num_cta = num_cta);
SPOOL OFF

```

Programa 3.1 NUM_ALU.SQL

```

/*****
/* Programa que obtiene el numero de alumnos */
/* de una generacion dada inscriros al semestre */
/* actual */
/* */
/* SMOD:          ACABAS */
/* */
/* Entradas:      generacion */
/* */
/* Archivos consultados: ALUMNO */
/*                  INSCRITO */
/* Programa:      NUM_ALU.NAT */
/* */
*****/

```

```

Define Report 1 'rep$tesis:num_alu'
Reset #sum (n4)
Input 'Generacion (999) :' #gen (n3)
L1. Find ALUMNO with per_ing_fac = #gen
L2. Find INSCRITO with num_eta = num_eta (L1.)
   Add 1 to #sum
   Loop
Loop
Display (1) #sum
End

```

Programa 3.2 NUM_ALU.NAT

SMBD: ORACLE

Num. Procesos usando SMBD	Num. E/S	Num. Faltas Pagina	Tiempo Procesador	Tiempo Total
1	1098	1986	01:42.12	02:54.82
3	1855	2041	01:49.97	07:09.08
6	1541	1983	01:50.47	12:59.95
9	2275	1982	01:57.95	20:23.13
12	2151	1982	01:59.03	27:21.82
15	2053	1955	02:05.77	35:22.58

SMBD: ADABAS

Num. Procesos usando SMBD	Num. E/S	Num. Faltas Pagina	Tiempo Procesador	Tiempo Total
1	117	1431	00:31.74	01:50.52
3	121	1440	00:31.51	05:21.50
6	118	1445	00:32.88	10:59.79
9	127	1447	00:33.36	15:24.95
12	121	1447	00:32.29	20:07.80
15	125	1451	00:32.43	24:57.78

TABLA 3.4 Estadísticas de la aplicación NUM_ALU.

Selección de un Manejador de Bases de Datos

```
REM /*****-----***/
REM /* Programa que obtiene el numero de alumnos */
REM /* inscritos a cada materia */
REM /* */
REM /* SMBO: ORACLE */
REM /* */
REM /* Entradas: generacion */
REM /* */
REM /* Tablas consultadas: ALUMNO */
REM /* INSCRITO */
REM /* */
REM /* Programa: ALU_MAT.SQL */
REM /* */
REM /*****-----***/

CLEAR COLUMN
COLUMN sysdate NEW_VALUE fecha
COLUMN sysdate NOPRINT
TTITLE RIGHT fecha -
SKIP CENTER 'NUMERO DE ALUMNOS INSCRITOS DEL SEMESTRE '&1 -
SKIP 2

SPOOL rep$tesis:sel_alu_mat
SELECT sysdate,materia "Materia",count(*) "Numero de alumnos
inscritos"
FROM cl_inscrito
WHERE num_cia IN
(SELECT num_cia
FROM cl_alumno
WHERE per_ing_fac = &1)
GROUP BY materia;
SPOOL OFF
```

Programa 3.3 ALU_MAT.SQL

```

/*****
/* Programa que obtiene el número de alumnos */
/* inscritos a cada materia */
/* */
/* SMED: ADABAS */
/* */
/* Entradas: generacion */
/* */
/* Archivos consultados ALUMNO */
/* INSCRITO */
/* */
/* Programa: ALU_MAT.NAT */
/* */
*****/

Define report 1 'rep$tesis:alu_mat.lis'
Reset #vec_mat (n3/1:999)
Input 'Generacion (???) :' #gen (n3)
L1. Find ALUMNO with per_ing_fac = #gen
L2. Find INSCRITO with num_cia = num_cia (L1.)
    For #i 1 c#mat_gru
        move materia( #i) to #mat
        Add 1 to #vec_mat ( #mat )
    Loop
Loop
Loop

display (1) 'Materias Numero de alumnos'
For #i 1 999
    If #vec_mat ( #i ) > 0 do
        Display (1) #i #vec_mat ( #i )
    Doend
Loop
End

```

Programa 3.4 ALU_MAT.NAT

SMBD: ORACLE

Num. Procesos usando SMBD	Num. E/S	Num. Falta Pagina	Tiempo Procesador	Tiempo Total
1	1861	2325	02:07.01	03:10.04
3	2043	2335	02:13.38	03:26.64
6	2113	2307	02:15.32	15:47.84
9	1724	2306	02:18.98	23:20.47
12	2806	2315	02:27.05	33:00.70
15	2607	2309	02:29.84	42:32.87

SMBD: ADABAS

Num. Procesos usando DBMS	Num. E/S	Num. Falta Pagina	Tiempo Procesador	Tiempo Total
1	130	1198	01:26.15	03:28.70
3	130	1459	01:25.76	09:10.57
6	117	1457	01:24.77	18:55.74
9	121	1462	01:25.14	28:58.42
12	126	1331	01:26.26	44:30.10
15	120	1475	01:27.20	55:56.57

Tabla 3.5 Estadísticas de la aplicación ALU_MAT.

```

REM      /******
REM      /* Programa que obtiene el numero de alumnos      */
REM      /* inscritos a 0,1...7 materias de alguna generacion*/
REM      /*                                                */
REM      /* SMDB:                ORACLE                    */
REM      /*                                                */
REM      /* Entradas:            generacion                */
REM      /*                                                */
REM      /* Tablas consultadas: ALUMNO                    */
REM      /*                      INSCRITO                  */
REM      /*                                                */
REM      /* Programa:            MATO-7.SQL                */
REM      /*                                                */
REM      /******

```

REM Se genera una vista que contiene por cada alumno el
 REM numero de materias a las que esta inscrito.

```

CREATE VIEW num_materias (num_ota,num_mat)
AS SELECT el_alumno.num_ota, count(el_inscrito.num_ota)
   FROM el_inscrito,el_alumno
  WHERE el_inscrito.num_ota (+) = el_alumno.num_ota
     AND per_ing_fac = &1
  GROUP BY el_alumno.num_ota;

```

```

CLEAR COLUMN
COLUMN sysdate NEW_VALUE fecha
COLUMN sysdate HOPRINT
TTITLE RIGHT fecha -
SKIP CENTER 'ALUMNOS INSCRITOS 'SKIP CENTER 'SEMESTRE '&1 -
SKIP 2

```

REM Se suman los alumnos que llevaron 0 materias,
 REM 1 materia, 2 materias ... 7 materias.

```

SPOOL rep$tesis:sal_alu_mato-7
SELECT sysdate,num_mat "Numero de Materias" ,
       count(*) "Numero de Alumnos"
FROM   num_materias
GROUP BY num_mat;
SPOOL OFF

```


Selección de un Manejador de Bases de Datos

```

/*****
/* Programa que obtiene el numero de alumnos
/* inscritos a 0,1...7 materias de alguna generacion */
/*
/* SMBO: ADABAS */
/*
/* Entradas: generacion */
/*
/* Archivos consultados: ALUMNO */
/* INSCRITO */
/* Programa: MATO-7.NAT */
*****/

Define report 1 'rep$tesis:mato-7.iis'
Reset #i (n1)
Reset #vector (n3/1:8)
Input 'Generacion (999) :' #gen (n3)
L1. Find ALUMNO with per_ing_fac = #gen
L2. Find INSCRITO with num_cia = num_cia (L1.)
If no records found do
Add 1 to vector (1)
escape
doend
move c$mat_gru to #i
Add 1 to #i
Add 1 to vector ( #i)
Loop
Display (1) nohdr vert as captioned
'numero de alumnos inscritos a 1 materias' '=' vector(1)
'numero de alumnos inscritos a 2 materias' '=' vector(2)
'numero de alumnos inscritos a 3 materias' '=' vector(3)
'numero de alumnos inscritos a 4 materias' '=' vector(4)
'numero de alumnos inscritos a 5 materias' '=' vector(5)
'numero de alumnos inscritos a 6 materias' '=' vector(6)
'numero de alumnos inscritos a 7 materias' '=' vector(7)
'numero de alumnos inscritos a 0 materias' '=' vector(8)
End

```

Programa 3.6 MATO-7.NAT

SMBD: ORACLE

Num. Procesos usando SMBD	Num. E/S	Num. Faltas Pagina	Tiempo Procesador	Tiempo Total
1	821	4690	01:44.96	02:45.34
3	2299	2454	01:48.29	06:59.62
6	1730	2347	01:48.37	12:41.01
9	1404	2341	01:50.06	19:04.05
12	2801	2324	01:59.18	25:25.94
15	2692	2340	02:05.58	32:30.85

SMBD: ADABAS

Num. Procesos usando SMBD	Num. I/O	Num. Faltas Pagina	Tiempo Procesador	Tiempo Total
1	116	1437	00:37.63	02:18.50
3	120	1456	00:37.91	06:23.54
6	122	1457	00:38.43	12:50.72
9	119	1452	00:38.29	19:45.36
12	124	1451	00:38.32	27:37.53
15	121	1456	00:38.23	33:30.23

Tabla 3.6 Estadísticas de la aplicación MAT 0-7.

3.4 Conclusiones

Después de haber realizado el estudio teórico anterior, así como las pruebas experimentales pertinentes, se presentan a continuación los puntos de análisis más importantes que permitirán hacer la elección del manejador de bases de datos.

TOPICO	ORACLE	ADABAS
Administración y recuperación del espacio en disco	El tiempo requerido para borrar la información es considerable. No es posible recuperar espacio apartado a una tabla hasta que ésta es borrada.	La base de datos debe estar cargada en el mismo lugar donde fue instalada, dificultando con esto la compactación del disco.
Flexibilidad de la estructura de la base de datos	Es muy flexible. Es posible alterar la estructura sin necesidad de vaciar la información.	Es muy flexible. Es posible alterar la estructura sin necesidad de vaciar la información.
Facilidades del lenguaje de manipulación de datos	SQL tiende a ser el lenguaje estándar para el manejo de bases de datos y por lo tanto tiene una gran difusión. Es sencillo y poderoso.	NATURAL es un lenguaje propio de ADABAS. Cuenta con algunas características de un lenguaje de tercera generación.
Productos adicionales	ORACLE cuenta actualmente con un conjunto completo de productos que ORACLE provee para hacer más eficiente y productiva la explotación de la base de datos.	Aunque el ambiente ADABAS cuenta con una serie de programas que giran en torno a la base de datos, ORACLE no dispone actualmente de ellos.
Administración	La administración de la base de datos es eficiente y sencilla. No requiere de muchos conocimientos.	El ambiente de administración no es muy amigable. Ciertos procesos requieren un conocimiento un poco más profundo.

TOPICO	ORACLE	ADABAS
Seguridad	El esquema de seguridad es eficiente, semejante al implementado en VME y muy adecuado a las necesidades del CECAFI.	CECAFI no cuenta actualmente con la utilidad RACADM SECURITY por lo que su sistema de seguridad no es muy poderoso.
Documentación	Es completa, didáctica y organizada.	No es muy didáctica.
Proceso de carga de información	Es un proceso muy sencillo aunque consume mucho tiempo de máquina.	Es un proceso poco amigable que implica adecuar los datos de entrada, comprimir la información, hacer cálculos del espacio que ocupará y cargar la información. Sin embargo el tiempo de máquina empleado es notablemente pequeño.
Cantidad de espacio requerido	La cantidad de espacio puede aumentar más del 100% respecto a los datos originales.	Aunque depende de la cantidad y tipo de información generalmente aumenta menos del 100% respecto a los datos originales.
Tiempo de acceso	El tiempo de acceso a los datos es aceptable, si se emplean los métodos adecuados. Sin embargo es mayor que el empleado por ADABAS.	El tiempo de acceso a los datos y en general el empleado por cualquier operación es altamente satisfactorio.

Del estudio anterior se aprecia que ADABAS tiene como principales virtudes un superior tiempo de respuesta y un manejo eficiente del espacio en disco, con el gran inconveniente en este aspecto que la base de datos debe estar localizada en las mismas direcciones físicas en que fue instalada, dificultando con esto el proceso de reubicación de archivos que constantemente se realiza en ambientes como el del CECAFI donde los discos se fragmentan rápidamente.

ORACLE tiene como principales ventajas contar con un lenguaje como SQL que es muy poderoso y de gran empuje en la actualidad, tener "software" de cuarta generación que disminuye en gran parte el tiempo de desarrollo de sistemas, contar con un ambiente de administración muy fácil de utilizar y ofrecer

una mayor portabilidad y conectividad que permite distribuir la carga de trabajo entre diversos sistemas.

Aunque el tiempo de respuesta es un factor importante a considerar y en este aspecto ADABAS es superior, pensamos que la diferencia no es notable y podría ser mejorada en cuanto se tengan discos más rápidos. Sin embargo, las herramientas de desarrollo de ORACLE, todas girando en torno al lenguaje estándar SQL, su ambiente de administración y la gran conectividad que tiene, lo que permitiría en un futuro tener computadoras personales de desarrollo conectadas a la computadora central o incluso tener bases de datos distribuidas, son factores que inclinan la balanza a su favor.

Después de este análisis, podemos afirmar que el CECAFI cuenta con dos manejadores de bases de datos de excelente calidad que debe aprovechar para difundir el uso de éstas en la Facultad. Para el desarrollo de sistemas y como plataforma para la implantación de nuestro modelo, seleccionamos a ORACLE por considerarlo un manejador de bases de datos ideal para el ambiente y necesidades del Centro.

Implantación

La creación del esquema es un proceso que tiene como objetivo la instalación física de las tablas definidas en la fase de modelado. En esta fase del proyecto fue necesario considerar las características fundamentales del manejador de bases de datos ORACLE para lograr que esta implantación fuera lo más eficiente posible. A continuación se presentan algunos conceptos de ORACLE que es necesario conocer para implantar el esquema, así como la manera en que se realizó este proceso. Un objetivo importante en este punto fue dejar establecidos los procedimientos necesarios para el cargado de las tablas, de tal manera que en el futuro sea fácil hacer modificaciones a este esquema.

4.1 Consideraciones Físicas

4.1.1 Particiones y archivos

La base de datos está constituida por unidades lógicas llamadas particiones, las cuales están asociadas a uno o más archivos físicos. Existe una partición llamada SYSTEM que se genera desde el momento de la inicialización de la base de datos y en

ella se carga el diccionario de datos y las tablas de usuarios que no se crean explícitamente en otra partición.

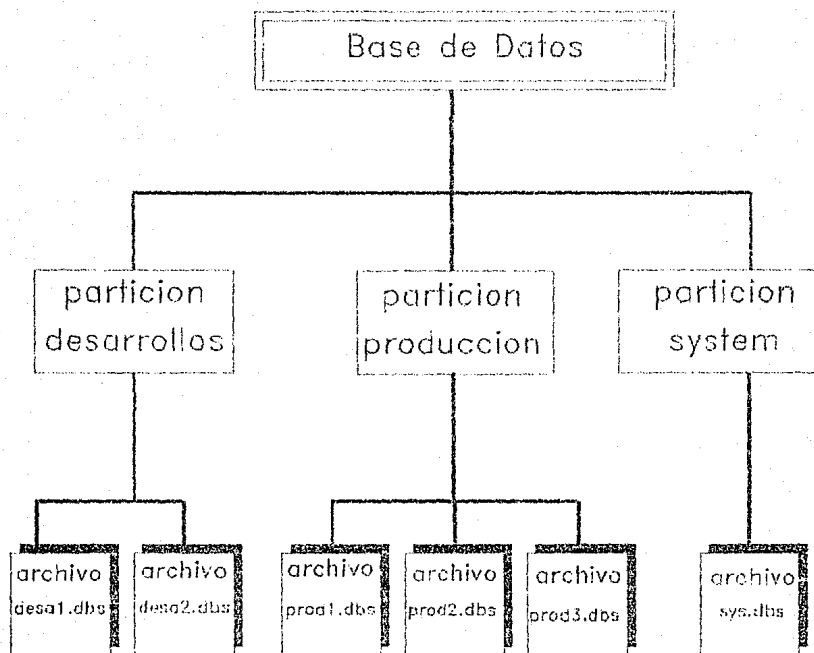


Figura 4.1 Particiones y archivos.

Cuando se estima el tamaño de una partición, se debe considerar suficiente espacio para todas las tablas que va a albergar. En caso de agotarse el espacio asignado a una partición, no podrán generarse más tablas ni podrán crecer las existentes; para remediar la situación debe asignarse otro archivo físico a dicha partición.

Las particiones se crean y modifican con el comando CREATE PARTITION y ALTER PARTITION. El espacio contiguo para los archivos se obtiene con la utilería CCF (Create Contiguous File).

El espacio que actualmente se encuentra libre en cada partición se puede obtener con la consulta:

```
SELECT SPM$PID, SPM$ENDBLOCK - SPM$STARTBLOCK + 1
FROM   SYS.SPACEMAP;
```

Es preferible contar con pocos conjuntos de muchos bloques, que el tener muchos de pocos bloques, en donde aumenta la posibilidad de no contar con suficiente espacio en disco que satisfaga una demanda. A esta última se lo llama una base de datos fragmentada e indica la necesidad de reorganizarla. La reorganización se puede lograr utilizando las utilerías EXPORT/IMPORT.

La siguiente es una consulta muy útil, ya que proporciona los valores promedio, mínimo y máximo de bloques contiguos en una partición.

```
SELECT AVG(SPM$ENDBLOCK-SPM$STARTBLOCK + 1) "PROMEDIO",
       MIN(SPM$ENDBLOCK-SPM$STARTBLOCK + 1) "MINIMO",
       MAX(SPM$ENDBLOCK-SPM$STARTBLOCK + 1) "MAXIMO",
       COUNT(*) "CUANTO", SPM$PID "PARTICION"
FROM   SYS.SPACEMAP
GROUP BY SPM$PID;
```

Cuando los números se tornan bajos, se plantea la posibilidad de agregar otro archivo en la partición requerida.

4.1.2 Tablas, segmentos y extensiones

Las tablas son la unidad básica de almacenamiento. Tienen todas las características de una tabla de Base de Datos Relacional. Las tablas están constituidas por dos segmentos: un segmento de datos y un segmento de índices. Este último existe aunque no se haya generado ningún índice para la tabla. Cuando una tabla es creada, se reserva el espacio contiguo requerido en la partición para los futuros datos e índices, estos bloques contiguos se conocen como "extents" o extensiones. Cuando una tabla crece ya sea en datos o en índices se aparta una nueva extensión en el segmento requerido. La primera extensión dada a una tabla en cada segmento en el momento de la creación se conoce como extensión inicial. El resto de la extensiones son conocidas como incrementales.

ESPACIO DE UNA TABLA

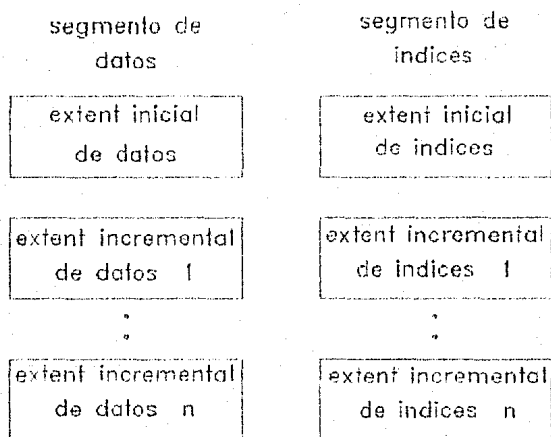


Figura 4.2 Tablas, segmentos y extensiones.

Para crear exitosamente una tabla, al menos debe existir en la partición un conjunto de bloques contiguos libres de tamaño mayor o igual al tamaño de las extensiones iniciales. En forma similar, para que una tabla crezca y adquiera una nueva extensión, deben existir suficientes bloques contiguos para formar la nueva extensión. Es conveniente además, tomar en cuenta que el espacio asignado a una tabla (en cualquiera de sus segmentos) no se libera para ser ocupado por otras tablas hasta que ésta es borrada (comando DROP).

4.1.3 Formato del bloque ORACLE

El tamaño de un bloque ORACLE depende del Sistema Operativo. Para la VAX es de 2048 bytes. No todo el espacio del bloque se utiliza para almacenar datos, ya que toda tabla requiere de un espacio adicional ("overhead") para guardar información útil para el SMBD. Cada bloque requiere de 76 bytes, que son utilizados para los encabezados del bloque físico de datos --44 bytes-- y del bloque lógico¹ de datos --32 bytes-- de tal suerte que de cada bloque restan 1972 bytes. Cada tabla, además, requiere dos bloques de datos y uno de índices para almacenar información de control; parte de esta información duplica algo del diccionario de datos para evitar su constante consulta. Cada renglón requiere de 4 bytes, para tablas no agrupadas, y 5 bytes para las agrupadas. Finalmente, cada columna ocupa dos bytes adicionales y esto se repite para cada renglón.

¹ El concepto de bloque lógico sólo aplica cuando se trata de un tabla agrupada ("cluster"); en este caso un bloque físico ORACLE tiene varios bloques lógicos ORACLE. Para las tablas no agrupadas cada bloque físico tiene un bloque lógico.

4.1.4 Definición de espacio

Para permitir controlar el crecimiento de una tabla, existe el concepto de definición de espacio. Mediante la definición de espacio se indica el número de bloques que tendrán las extensiones iniciales e incrementales (en ambos segmentos), así como el número máximo de extensiones permisible y el porcentaje de espacio libre dejado en cada bloque.

Existe una definición de espacio que se utiliza predefinida que se utiliza cuando se crea una tabla, pero el usuario puede crear sus propias definiciones. La definición de espacio **no aparta espacio**, sólo indica las características que tendrá la tabla cuando sea creada. Para asociar una tabla y una definición de espacio, esta última se nombra en la proposición CREATE TABLE. Ejemplo:

```
CREATE TABLE MITABLITA (SERIE NUMBER, NOMBRE CHARACTER)
SPACE ESPACIO;
```

en donde ESPACIO es una definición previa de espacio.

Una creación de espacio sería como:

```
CREATE SPACE DEFINITION ESPACIO
DATAPAGES (INITIAL 5,
           INCREMENT 25,
           MAXEXTENTS 9999,
           PCTFREE 20)
INDEXPAGES (INITIAL 5,
            INCREMENT 25,
            MAXEXTENTS 9999)
PARTITION SYSTEM;
```

La definición de espacio indica valores tanto para el segmento de datos como para el segmento de índices (DATA/PAGES e INDEX/PAGES respectivamente). Habla de un valor INITIAL, que define la cantidad de bloques que tendrá la extensión inicial. Estos bloques se encontrarán contiguos. Es deseable que la totalidad de la tabla se encuentre en bloques contiguos.

El valor de incremento, INCREMENT, se refiere al número de bloques contiguos que se tomarán cada vez que el espacio de la tabla se termine (extensiones incrementales).

MAXEXTENTS limita la cantidad de extensiones que se pueden asignar a la tabla. Usualmente es un número muy grande.

PCTFREE o porcentaje libre de bloque, es un valor muy importante. Se refiere al espacio que se dejará libre en cada bloque para futuras actualizaciones de los valores en los tuplos. Depende mucho de la inestabilidad de los datos de cada tabla, como cuando se carga información inicial con valores nulos, la que posteriormente se va a completar. El hecho de dejar una parte del bloque vacía evita que surjan encadenamientos cuando se agrega la información faltante. Un PCTFREE de 1% reflejará que se espera una estabilidad muy grande, mientras que un valor de 9% denota que aún falta mucha información por obtener y/o modificar.

Con PARTITION se define alguna partición alterna en que será creada la tabla.

Se pueden hacer algunas recomendaciones para usar las definiciones de espacio:

- # Se prefiere contar con pocas extensiones de gran tamaño, que muchas pequeñas. Optimamente, todos los datos de una tabla deben estar en el primer conjunto de bloques contiguos.
- # Las actualizaciones pueden hacer crecer a los renglones individualmente, como cuando se insertan renglones con valores NULOS y posteriormente éstos son llenados. En este caso, se debe incrementar el valor de PCIFREE.
- # A menos de que realmente se deba fijar un límite a la cantidad de espacio de una tabla, al parámetro de MAXEXTENTS debe asignarse el mayor valor (9999), que indica el máximo valor permitido dependiendo del sistema.
- # Hay que recordar que debe haber suficientes bloques lógicos contiguos para formar la extensión inicial de datos y de índices. Así también se requieren suficientes bloques contiguos para las extensiones incrementales.

4.1.5 Tipos de datos en ORACLE

Al estimar el tamaño de la base de datos, y en especial del espacio ocupado por cada tabla, es necesario conocer el tamaño en bytes de cada tipo de datos para definir la longitud de cada atributo o columna. Los tipos básicos de datos ORACLE son:

CHAR. Cadena de caracteres de longitud variable hasta 240. No almacena blancos al final de la columna.

NUMBER. Puede almacenar virtualmente cualquier magnitud con una precisión de hasta 38 dígitos. El formato es de longitud variable, de tal forma que cualquier número utilice al menos 2 bytes: uno para signo y exponento, y otro por cada pareja de dígitos almacenados. De esta manera,

9	se almacena en 2 bytes,
99	se almacena en 2 bytes,
999	se almacena en 3 bytes,
9999	se almacena en 3 bytes,
99999	se almacena en 4 bytes,
etcétera.	

Las parejas de ceros a la derecha no son almacenadas.

DATE. Almacena valores de tipo fecha, que incluyen siglo, año, mes, día, hora, minutos y segundos; utiliza 7 bytes.

LONG. Cadena de caracteres semejante a CHAR, que puede almacenar hasta 65,536 de éstos.

En base al Modelo de Datos y al número de bytes ocupado por cada tipo de datos, se puede calcular el espacio requerido por cada tabla.

4.1.6 Cálculo del número de bloques para el segmento de datos

Antes de definir y hacer la carga inicial de las tablas de la base de datos, se debe aproximar hasta donde sea posible la cantidad de espacio que las tablas consumirán, para tratar de mantener la información junta y hacer accesos más eficientes.

Sin embargo, la versatilidad del manejador de la base de datos y del modelo de datos normalizado, permiten modificar, afinar y mejorar dinámicamente el esquema conceptual de la base de datos, por lo que, si en la aplicación real este esquema no se ajusta a las necesidades presentadas, se puede cambiar y adaptar como sea necesario.

Para calcular el número de bloques que ocupará inicialmente una tabla, es necesario conocer los siguientes parámetros:

- R el número esperado de renglones.
- C_i la longitud promedio de cada columna.
- N_i el porcentaje de ocurrencias nulas para cada columna.
- F porcentaje libre en cada bloque para futuras actualizaciones (también conocido como PCTFREE).

Además, para hacer el cálculo adecuado, hay que tomar en cuenta tanto el tamaño del bloque ORACLE, así como los bytes que ORACLE necesita para almacenar información de control, así pues se tiene:

- K indicador de longitud de columna, 1 en VAX.
- I número de identificación de columna, 1 en VAX.
- L longitud de renglón, 2 en VAX.
- Q número de secuencia de renglón, 2 en VAX.
- S tamaño del bloque lógico, 2048 en VAX.
- H encabezado de información por bloque, 76 en VAX.

Con base en estos parámetros es posible evaluar las siguientes fórmulas:

$$BR = \sum_i [(1-N_i) * (C_i+K+1)] + L + Q$$

$$RB = \text{Parte entera de } \left\lfloor \frac{(1-F) * (R-H)}{BR} \right\rfloor$$

$$Ed = \frac{R}{RB}$$

En donde:

BR es el número promedio de bytes por renglón;

RB es el número promedio de renglones por bloque;

Bd es la cantidad de bloques requeridos por ORACLE para la tabla.

Si estas fórmulas se particularizan para VAX tenemos:

$$BR = \text{SUM}_i [(1-N_i) * (C_i+2)] + 4$$

$$RB = \text{Parte entera de } \left\lfloor \frac{(1-F) * 1972}{BR} \right\rfloor$$

$$Ed = \frac{R}{RB}$$

4.1.7 Cálculo del número de bloques para el segmento de índices

Tal como se mencionará posteriormente, el uso de mecanismos de indexación permite lograr una mayor eficiencia en el desempeño de la base de datos, aunque consume grandes volúmenes de espacio. El espacio requerido para índices es difícil de precisar, éste depende de factores como:

- Los valores diferentes que puede tomar una columna de índice y qué tan diferente es uno de otro,

Implantación

- el número de columnas indexadas,
- el porcentaje de NULDS,
- si el índice se comprime.

El espacio puede estimarse con la fórmula:

$$Bi = 1.1 * R / (((1-F)*(S-H))/KL)$$

en donde:

- R número de renglones en la tabla.
- S tamaño de bloque (2048 en VAX).
- H encabezado por bloque (usualmente 76 bytes).
- F porcentaje libre que se deja para actualizaciones (PCTFREE).
- KL longitud estimada de la llave.
 - 10 para índices comprimidos.
 - 16 + SLL + CL, para índices no comprimidos.
- CL número de columnas en la llave.
- SLL suma de longitudes de columnas en la llave.

Usualmente será requerido el espacio para índices, puesto que estos son necesarios para acelerar los procesos de la base de datos y para asegurar la validez de los datos, al controlar las no repeticiones de las llaves.

4.2 Identificación de tablas y columnas

A pesar de la claridad con que los nombres de Entidades, Asociaciones y Atributos representan a la realidad, se debió otorgar otro nombre a las tablas y columnas asociadas para poder cumplir con las reglas sintácticas establecidas por el SMOB ORACLE. Estos nombres se escogieron los más semejantes posible a las Entidades, Asociaciones y Atributos que les dieron origen.

En la tabla 4.1 se resumen los nombres de Entidades y Asociaciones, así como las tablas respectivas.

En la tabla 4.2 se muestran los nombres de las columnas asociadas con cada atributo, incluyendo además su tipo y longitud en bytes para poder calcular el espacio que consumirá en la base de datos.

Nombre de la entidad	Nombre de la tabla en el esquema
ACADENICO	ACADENICO
ACTUALIZACION	ACTUALIZACION
ALUMNO	ALUMNO
ALUMNO EXTERNO	ALUMNO_EXTERNO
ALUMNO INTERNO	ALUMNO_INTERNO
ALUMNO NUEVO INGRESO	ALUMNO_NUEVO_INGRESO
ASIGNATURA	ASIGNATURA
CARRERA	CARRERA
DEPARTAMENTO	DEPARTAMENTO
DIA	DIA
EDIFICIO	EDIFICIO
GRUPO	GRUPO
GRUPO EXTRA	GRUPO_EXTRA
GRUPO TUTOR	GRUPO_TUTOR
LUGAR DE ADEUDO	LUGAR_DE_ADEUDO
NOMINA	NOMINA
ORGANO	ORGANO
PROCEDENCIA	PROCEDENCIA
SALON	SALON
TESIS	TESIS

Tabla 4.1 Resumen de Tablas de Entidades y Asociaciones.

Nombre de la Asociación	Nombre de la tabla en el esquema
Adeuda	ADEUDA
Alta baja cambio	ABC
Cita	CITA
Clase	CLASE
Compone	COMPONE
Contrata	CONTRATA
Equivalencia	EQUIVALENCIA
Estudia	ESTUDIA
Examen	EXAMEN
Falta	FALTA
Ha estudiado	HA_ESTUDIADO
Inscrito	INSCRITO
Inscrito extra	INSCRITO_EXTRA
Paga	PAGA
Presentó	PRESENTO
Seriación	SERIACION
Sinodal	SINODAL
Solicita	SOLICITA
Tutoria	TUTORIA

Table 4.1 (Continuación).

Nombre del atributo en el modelo	Nombre del atributo en esquema	Tipo	Long
año de alta administrativos	ANO_ALTA_ADM	CH	2
año de ingreso a DGAE	INGRESO_DGAE	CH	2
año de inicio tutoría	INICIO	CH	2
año-semestre de evaluación en H.	EVALUACION_HA	CH	3
año-semestre de ingreso a la Facultad	INGRESO_FACULTAD	CH	3
año-semestre de inicio	INICIO	CH	3
año-semestre de registro a carrera	REGISTRO	CH	3
año-semestre presentó último extraordinario	ULTIMO_EXTRA	CH	3
años de servicio académico	SERVICIO	N(2)	3
asignaturas aprobadas en extraordinario	ASIGNATURAS_EXTRA	N(3)	3
asignaturas aprobadas en ordinario	ASIGNATURAS_ORDINARIO	N(3)	3
asignaturas reprobadas	REPROBADAS	N(3)	3
calificación	CALIFICACION	CH	1
capacidad	CAPACIDAD	N(5)	3
clave de asignatura	CLAVE, ASIGNATURA	N(4)	3
clave de asignatura actual	ACTUAL	N(4)	3
clave de asignatura antigua	ANTIGUA	N(4)	3
clave de asignatura antecedente	ANTECEDENTE	N(4)	3

Tabla 4.2 Resumen de Atributos.

Nombre del atributo en el modelo	Nombre del atributo en esquemas	Tipo	Long
clave de asignatura consecuente	CONSECUENTE	N(4)	3
clave de categoría	CLAVE, CLAVE_CATEGORIA	CH	4
clave de departamento	CLAVE, DEPARTAMENTO	N(2)	2
clave de edificio	CLAVE, EDIFICIO	N(2)	2
clave de la carrera	CLAVE, CARRERA	N(2)	2
clave de lugar de adeudo	CLAVE	N(2)	2
clave de procedencia	CLAVE, NACIONALIDAD	N(3)	3
clave de tesis	CLAVE	N(6)	4
clave del estado	ESTADO	N(1)	2
clave del órgano	CLAVE, ORGANO	N(2)	2
clave desconocida	BESCONOCIDA	CH	3
clave escuela de procedencia	PROCEDENCIA	N(3)	3
código postal	CP	N(5)	4
colonia/población	COLONIA	CH	30
créditos	CREDITOS	N(2)	2
créditos actuales	CREDITOS	N(3)	3
créditos cubiertos	CREDITOS_CUBIERTOS	N(3)	3
créditos cursando	CREDITOS	N(2)	2
créditos revalidados	CREDITOS_REVALIDADOS	N(3)	3
cupo	CUPO	N(2)	2

Tabla 4.2 (Continuación).

Nombre del atributo en el modelo	Nombre del atributo en esquema	Tipo	Long
delegación/municipio	DELEGACION	CH	20
dependencia-subdependencia	DEPENDENCIA	CH	5
dígito de control	CONTROL	CH	2
domicilio	DOMICILIO	CH	37
estado de contratación	ESTADO	CH	1
fecha de actualización alumnos	ALUMNOS	CH	3
fecha de adeudo	FECHA	D	7
fecha de alta académicos	ALTA_ACAD	D	7
fecha de examen profesional	FECHA	D	7
fecha de falta	FECHA	D	7
fecha de inicio de servicio académico	INICIO_SERVICIO	D	7
fecha de la nómina	NOMINA	D	7
fecha de nacimiento	NACIMIENTO	D	7
fecha límite de contrato académico	LIMITE	D	7
fecha y hora de extraordinario	FECHA	D	7
folio	FOLIO	N(6)	4
folio del grupo	FOLIO	CH	11
grupo primer ingreso	PRIMER_GRUPO	N(2)	2
grupo tutor	GRUPO	N(2)	2

Tabla 4.2 (Continuación).

Nombre del atributo en el modelo	Nombre del atributo en esquema	Tipo	Long
hora final	HORA_FINAL	N(4)	3
hora final de asesoría	HORA_FINAL	N(4)	3
hora inicial	HORA_INICIAL	N(4)	3
hora inicial de asesoría	HORA_INICIAL	N(4)	3
horas académicos	HORAS	N(3.1)	3
índice de escolaridad	ESCOLARIDAD	N(3)	3
índice de velocidad	VELOCIDAD	N(3)	3
inscripciones en ordinario	INSCRIPCIONES	N(3)	3
irregularidad	IRREGULARIDAD	CH	1
laboratorio o práctica	LABORATORIO	CH	1
lugar de tutoría	LUGAR	CH	126
mes alta administrativos	MES_ALTA_ADM	CH	1
motivo de no inscrito	MOTIVO	CH	1
movimiento	MOVIMIENTO	CH	1
nivel	NIVEL	CH	2
nombre de categoría	NOMBRE, NOMBRE_CATEGORIA	CH	14
nombre de la asignatura	NOMBRE	CH	28
nombre de la carrera	NOMBRE	CH	34
nombre de la tesis	NOMBRE	CH	64
nombre de lugar de aduado	NOMBRE	CH	64

Tabla 4.2 (Continuación).

Nombre del atributo en el modelo	Nombre del atributo en esquema	Tipo	Long
nombre de órgano	NOMBRE	CH	53
nombre de procedencia	NOMBRE	CH	21
nombre del académico	NOMBRE	CH	52
nombre del alumno	NOMBRE	CH	32
nombre del departamento	NOMBRE	CH	52
nombre del día	NOMBRE	CH	9
nombre en nómina	NOMBRE	CH	25
número de cuenta	CUENTA	N(8)	5
número de departamento	NUMERO	N(2)	2
número de día	NUMERO, DIA	CH	1
número de examen	NUMERO	N(5)	3
número de grupo	NUMERO	N(2)	2
número de grupo extraordinario	NUMERO	N(2)	2
número de grupo tutor	NUMERO, NUMERO_GT	N(4)	3
número de laboratorios inscrito	LABORATORIOS	N(1)	2
número de movimientos en r.a.	MOVIMIENTOS_RA	N(5)	3
número de plaza	PLAZA	CH	3
número de profesor	NUMERO_PROFESOR	CH	1
número de salón	NUMERO, SALON	CH	4
número de sorteo	SORTEO	N(5)	4

Tabla 4.2 (Continuación).

Nombre del atributo en el modelo	Nombre del atributo en esquema	Tipo	Long
número de teorías inscrito	TEOR_AS	N(1)	2
número en nómina	NUMERO	N(4)	3
número único de grupo	NUMERO_UNICO	N(5)	4
número único de grupo alta	NUMERO_UNICO_ALTA	N(5)	4
número único de grupo baja	NUMERO_UNICO_BAJA	N(5)	4
número único de grupo extraordinario	NUMERO_UNICO	N(3)	5
obligatoriedad	OBLIGATORIEDAD	CH	1
opción	OPCION	CH	1
partida	PARTIDA	CH	3
periodo	PERIODO	CH	1
piso a nivel	PISO	CH	1
plan escolar asignado	PLAN	N(1)	2
porcentaje de avance	AVANCE	N(4.1)	3
posible consejero	CONSEJERO	CH	1
primer salón	PRIMER_SALON	CH	4
primero o segundo	PRIMERO_SEGUNDO	CH	1
programa-subprograma	PROGRAMA	CH	4
promedio	PROMEDIO	N(4.2)	3
proyección	PROYECCION	CH	1

Tabla 4.2 (Continuación).

Implantación

quincenas pagadas	PAGADAS	N(4)	4
registro federal de causante	RFC	CH	13
resultado	RESULTADO	CH	1
semestre	SEMESTRE	N(2)	2
semestre grupo	SEMESTRE	CH	1
semestres sin cursar asignatura	SIN_CURSAR	N(3)	3
sexo	SEXO	CH	1
sigla	SIGLA	CH	6
sueldo	SUELDO	N(9)	6
teléfono	TELEFONO	N(7)	5
tipo de alumno	TIPO	CH	1
tipo de clase	TIPO	CH	1
tipo de examen	TIPO	CH	1
tipo de falta	TIPO	CH	1
título	TITULO	CH	5
ubicación de edificio	UBICACION	CH	32
último año semestre en cursar asignatura	ULTIMO_EN_CURSAR	CH	3
vacantes	VACANTES	N(2)	2
vigencia	VIGENCIA	CH	1

Tabla 4.2 (Continuación).

En las tablas anteriores la columna Tipo indica uno de los posibles tipos de datos de ORACLE:

CH	character
D	date
L	long
N	number
N(n)	number (n dígitos)
N(n.d)	number (n total de dígitos, d decimales). El punto no se toma en cuenta.

La información que aparece en la columna Long es la longitud real que ocupa el campo: en character depende del número de caracteres, en date siempre es 7, en long depende del número de caracteres y en number depende indirectamente del número de dígitos.

4.3 Espacio estimado de la Base de Datos

Tomando datos estimados por los usuarios y considerando índices en toda llave primaria, se hicieron los cálculos necesarios para poder estimar el volumen de la base de datos. En las siguientes tablas se muestran los resultados. A los parámetros ya mencionados, se agregaron los siguientes:

Rc Número de renglones por cada incremento

Nc Número de columnas (atributos)

Bdc Cantidad de bloques de datos requeridos por la tabla en cada incremento

Bic Cantidad de bloques de índices requeridos por la tabla en cada incremento

TABLA	Ri	Rc*	Eci	F	SLL	CL	NC
ACADEMICO	800	50	55	0.05	15	1	5
ACTUALIZACION	1	0	10	0.00	0	0**	2
ALUMNO	12000	0	16	0.00	5	1	6
ALUMNO_EXTERNO	100	0	40	0.01	5	1	3
ALUMNO_INTERNO	20000	3000	192	0.05	5	1	28
ALUMNO_NUEVO_INGRESO	5000	0	150	0.05	5	1	15
ASIGNATURA	901	0	40	0.05	5	1	7
CARRERA	11	0	41	0.05	2	1	4
DEPARTAMENTO	30	0	38	0.05	2	1	3
DIA	7	0	10	0.00	1	1	2
EDIFICIO	40	0	54	0.00	2	1	2
GRUPO	3700	0	14	0.00	4	1	6
GRUPO_EXTRA	376	0	30	0.00	3	1	6
GRUPO_TUTOR	180	90	150	0.00	3	1	6
LUGAR_DE_ADEUDO	50	0	68	0.05	2	1	3
HONINA	3300	0	145	0.00	25	1	20
ORGANO	18	0	55	0.00	2	1	2
PROCEDENCIA	150	0	24	0.00	3	1	2
SALON	250	0	12	0.05	4	1	6
TESIS	2600	100	62	0.01	4	1	2

* Crecimiento semestral o anual.

** No requiere llave indexada.

Tabla 4.3 Parámetros estimados para las tablas de la Base de Datos (Entidades).

TABLA	Bd	Bdc	Bi	Bic
ACADEMICO	30	2	15	1
ACTUALIZACION	1	0	0	0
ALUMNO	197	0	148	0
ALUMNO EXTERNO	3	0	2	0
ALUMNO INTERNO	2858	429	259	39
ALUMNO NUEVO INGRESO	300	0	39	0
ASIGNATURA	29	0	11	0
CARRERA	1	0	1	0
DEPARTAMENTO	1	0	1	0
DIA	1	0	1	0
EDIFICIO	1	0	1	0
GRUPO	57	0	44	0
GRUPO EXTRA	7	0	5	0
GRUPO TUTOR	17	9	3	1
LUGAR DE ADEUDO	3	0	1	0
NOMINA	330	0	78	0
ORGANO	1	0	1	0
PROCEDENCIA	3	0	2	0
SALON	4	0	3	0
TESTS	105	4	31	2

SUMAS EN ENTIDADES	3949	444	646	43
--------------------	------	-----	-----	----

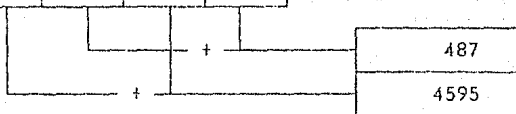


Tabla 4.4 Cálculo tentativo de bloques iniciales y de crecimiento para las tablas de la Base de Datos (Entidades).

Implantación

TABLA	Ri	Rc*	ECi	F	SLL	CL	NC
ABC	12000	0	15	0.00	7	2	5
ADEUDA	2000	500	14	0.00	7	2	3
CITA	180	90	10	0.00	4	2	4
CLASE	3963	0	30	0.00	22	4	8
COMPONE	411	0	7	0.00	5	2	4
CONTRATA	3148	314	15	0.00	15	2	2
EQUIVALENCIA	1802	0	6	0.00	6	2	2
ESTUDIA	20000	3000	13	0.00	7	2	4
EXAMEN	18824	650	20	0.00	9	2	5
FALTA	354	0	25	0.00	17	2	4
HA_ESTUDIADO	1600	100	14	0.00	7	2	5
INSCRITO	40000	3000	9	0.00	9	2	2
INSCRITO_EXTRA	4000	0	8	0.00	8	2	2
PAGA	3500	0	16	0.00	3	1	2
PRESENTO	1 000000	65000	22	0.00	19	3	6
SERIACION	1802	0	6	0.00	6	2	2
SINODAL	435	0	17	0.00	16	2	3
SOLICITA	15000	0	10	0.00	10	3	3
TUTORIA	6000	3000	8	0.20	8	2	2

* Crecimiento semestral o anual.

Tabla 4.5 Parámetros estimados de las tablas de la Base de Datos (Asociaciones).

TABLA	Bd	Bdc	Bi	Bic
ABC	177	0	181	0
ADEUDA	25	7	26	7
CITA	3	2	3	2
CLASE	102	0	93	0
COMPONE	4	0	6	0
CONTRATA	38	4	58	6
EQUIVALENCIA	13	0	25	0
ESTUDIA	257	39	279	42
EXAMEN	325	12	284	10
FALTA	7	0	5	0
HA_ESTUDIADO	23	2	23	2
INSCRITO	345	26	603	46
INSCRITO_EXTRA	33	0	59	0
PAGA	43	0	40	0
PRESENTO	19608	1375	21197	1378
SERIACION	13	0	25	0
SINODAL	6	0	9	0
SOLICITA	135	0	211	0
TUTORIA	62	31	109	55
SUMAS ASOCIACIONES	21217	1398	23238	1548

					2946
					44455

Tabla 4.6 Cálculo tentativo de bloques iniciales y de crecimiento para las tablas de la Base de datos (Asociaciones).

Sumando los bloques iniciales de datos requeridos para entidades y asociaciones y haciendo lo mismo para los índices, se tiene que se requiere aproximadamente de 25166 bloques iniciales para datos y 23884 bloques para índices. Recordando que un bloque de ORACLE es de 2048 bytes, se requiere de 49.15 Megabytes de espacio en disco para datos y 46.64 Megabytes para índices (aunque el número y tamaño de índices puede variar considerablemente).

Un detalle importante es el tamaño de la tabla PRESENTO, que por sí sola consume 19608 bloques para datos y 21197 para índices, que son 38.29 Megabytes de datos y 41.40 Megabytes para índices; restando los números anteriores del total calculado, se tiene que sólo se requieren al inicio 10.86 Megabytes para datos y 5.24 Megabytes para índices, para cargar la Base de Datos sin tener en disco a la tabla PRESENTO.

4.4 Creación de la Base de Datos

Para la creación de la base de datos, se consideraron las dos funciones principales que tiene el CEEAFI: el desarrollo y producción de los sistemas administrativos de la facultad por una parte y el apoyo en las actividades académicas de alumnos y profesores por la otra. Debido a esto se decidió instalar dos bases de datos independientes. La primera de ellas se identificó como CEEAFI y se creó con el objeto de albergar todo el esquema definido en este proyecto para el uso administrativo de la Facultad. La segunda se identificó como USUARIOS y tuvo como objetivo ser usada con fines únicamente académicos.

Es necesario resaltar que en el momento de la creación del esquema no se contó con todo el espacio en disco que se hubiera deseado, por lo que las bases de datos se crearon ocupando el espacio disponible. Esto no fue un gran inconveniente, pues la base de datos USUARIOS tuvo la suficiente capacidad para cumplir los objetivos para los cuales fue creada y la base de datos CECAFI se instaló del tamaño suficiente para alojar todas las tablas diseñadas, con la excepción de la tabla PRESENTO; la cual por su tamaño tendrá que ser cargada en cuanto el CECAFI cuente con unidades de disco de mayor capacidad. Esto no representará ningún problema, pues ORACLE permite agregar espacio a la base conforme la información va aumentando.

Los tamaños iniciales de las bases de datos, así como su nombre físico, fueron los siguientes:

```
CECAFI      75 MB   DISK$USUARIOS2:[ORACLE]CECAFI.DBS
USUARIOS    20 MB   DISK$USUARIOS2:[ORACLE]USUARIOS.DBS
```

El espacio fue creado contiguo mediante la utilería CCF de ORACLE con la opción CONTIG, para lograr así un mayor rendimiento del sistema, pues entre más cercana está la información los accesos a disco son más rápidos.

4.5 Definición de índices y agrupamiento de tablas

ORACLE proporciona dos medios para hacer más eficientes los accesos a las tablas: la creación de índices y el agrupamiento de tablas ("clusters"). Mediante los índices es posible tener acceso directo a un renglón en base al valor de un campo. Los

agrupamientos permiten optimizar los accesos que se hacen a los datos mediante "joins" de dos tablas, pues estas se almacenan en forma contigua; es decir, los renglones de dos tablas agrupadas mediante una columna común se encuentran en el mismo bloque lógico de ORACLE.

En este punto, se tuvo que determinar que campos indexar y qué tablas agrupar. Se decidió generar índices para todas las llaves primarias; estos índices se generaron con la opción UNIQUE para verificar la unicidad de renglones y sin comprimir, pues aunque el hacerlo ahorra espacio en disco, se pierde tiempo al hacer acceso a ellos. No se generaron índices de ningún otro campo, con la idea de hacerlo conforme los sistemas lo vayan requiriendo y no desperdiciar espacio en índices que no se usen adecuadamente.

La determinación de las agrupaciones no fue sencilla y su funcionalidad tendrá que evaluarse conforme los sistemas se desarrollen. Aunque las agrupaciones optimizan los "joins", pueden tener algunos efectos contraproducentes, sobre todo en cuanto a espacio en disco se refiere. Esto se debe a lo siguiente: como ya se ha comentado, los renglones de ambas tablas se agrupan bajo una columna común (llave del "cluster"), el factor importante a considerar es que en cada bloque lógico se almacena una sola llave y 300 campos asociados. Aunque es posible determinar el tamaño del bloque lógico, el menor para VMS es de 300 bytes, de tal forma que si existen ocurrencias con muy pocos campos para una llave, existirá un gran desperdicio de almacenamiento. Para decidir qué agrupaciones efectuar se tomó en cuenta que las parejas de tablas se usaran comúnmente en "joins", que tuvieran una cantidad semejante de campos

agrupados por cada llave y su permanencia en el disco fuera la misma. Así pues, se decidió formar las siguientes agrupaciones:

ALUMNO_INTERNO - ESTUDIA

GRUPO - CLASE

ORGANO - DEPARTAMENTO

El espacio ocupado por dos tablas agrupadas no puede calcularse mediante las fórmulas descritas anteriormente. Para hacer un cálculo aproximado debe tomarse en cuenta cuantas llaves diferentes habrá y cuantos bloques lógicos caben en un bloque físico. Hay que tomar en cuenta también que el espacio en disco no se administra por tabla, sino por todo el "cluster", es decir, hasta que sea borrado el "cluster" entero, se liberará el espacio ocupado.

4.6 Descripción general del proceso de carga

Estando conscientes del dinamismo de la información, se pensó en hacer los procedimientos de carga lo más estandarizados posible, para que en un futuro el administrador de la base de datos pudiera cargar cualquier tabla que fuera necesaria.

El primer paso fue la creación de una clave de ORACLE llamada ABD (Administrador de la Base de Datos). En esta clave privilegiada se crearon todas las tablas de la base de datos y mediante ésta el administrador dará los derechos de acceso que necesite cada usuario.

Implantación

Para el proceso de carga se creó una clave a nivel sistema operativo llamada BASE. El objetivo para la creación de esta clave fue tener en un mismo ambiente todos los procesos de carga, con sus respectivos archivos de entrada y salida, ya que actualmente la información está distribuida en muchos sistemas residentes en claves diferentes. En esta clave se realizaron todos los programas que ayudan en el proceso de carga de cualquier tabla. También se crearon directorios específicos para alojar los archivos de datos y de estadísticas. Para esto se definieron los siguientes nombres lógicos:

NOMBRE LOGICO	DIRECTORIO	CONTENIDO
Basefcar Disk[sistemas:[base.carga]		Programas fuente y ejecutables
Basefdat Disk[archivos:[archivos.base]		Archivos de datos de entrada
Basefstat Disk[sistemas:[base.estadisticas]		Archivos de entrada y salida para estadísticas
Basefgen Disk[sistemas:[base.genera]		Programas que generan los archivos de entrada
Basefrep Disk[archivos:[archivos.base.reportes]		Reportes de salida

A continuación se describe el proceso sistemático que tiene como fin la instalación de una tabla en la base de datos. En realidad el proceso es muy simple, los primeros pasos que se realizan son con el único objetivo de crear la tabla con la definición de espacio adecuada para que los datos e índices estén lo más contiguos posible.

4.6.1 Formación del archivo de datos.

En este paso es necesaria la cooperación del jefe de usuarios de la información a ser cargada. El producto final de este paso es la obtención de un archivo ASCII con toda la información que contendrá la tabla y exactamente con las mismas características que se especifican para dicha tabla en la definición del

esquema. Estas características incluyen longitud y tipo del campo, así como si tiene o no valores nulos. Se debe tomar en cuenta que los espacios en blanco serán considerados en el momento del cargado como valores nulos. La información debe estar filtrada y debe tenerse especial cuidado en validar llaves foráneas en las tablas (o archivos) donde sea llave primaria. Los archivos de datos se generan en el directorio base\$dat y tienen como nombre el de la tabla con extensión .dat. Los programas que los generan se encuentra en el directorio base\$gen.

4.6.2 Obtención de parámetros

El siguiente paso consiste en obtener una serie de parámetros del archivo anteriormente formado, los cuales servirán para poder calcular el espacio requerido por la tabla. Estos parámetros son el número de renglones (Ri), la longitud promedio de cada columna (Ci) y el porcentaje de valores nulos de cada una (Ni). Para la obtención de estos valores, se realizó un programa llamado ESTADISTICAG el cual recibe un archivo de entrada con el nombre <tabla>.ent que contiene las características del archivo a procesar y arroja dos archivos de salida: el primero llamado <tabla>.rep que es un reporte de los valores obtenidos y el segundo llamado <tabla>.est el cual contiene los mismos valores pero en el formato de entrada para el siguiente proceso.

4.6.3 Cálculo del espacio ocupado

Con los datos obtenidos en el paso anterior se debe calcular el espacio que ocupará la tabla en base a las fórmulas descritas anteriormente (siempre y cuando no estén agrupadas). Para esto se realizaron dos pequeños programas que calculan el número de bloques ORACLE requeridos para datos y para índices. Estos programas se llaman ESPACIO y ESPACIOI respectivamente. Cada uno de estos programas escribe sobre un archivo el número de bloques ORACLE necesarios. El programa ESPACIO requiere del archivo <tabla>.est generado anteriormente y su salida queda grabada en el archivo <tabla>.esp. A su vez, el programa ESPACIOI requiere un archivo con el nombre <tabla>.dfi (el cual hay que editar) y genera otro llamado <tabla>.ind.

4.6.4 Creación y cargado de tablas

El siguiente paso consiste en crear la definición de espacio, la tabla, sus índices y su sinónimo. Para sistematizar esto, se realizó un procedimiento de comandos llamado CREA.COM que requiere dos archivos: el primero debe llamarse <tabla>.sql y debe contener los comandos necesarios de SQL para definir el espacio que ocupará la tabla, crearla, crear sus índices y su sinónimo; el segundo debe contener las instrucciones propias de la utilería ODL para cargar la tabla con los datos del archivo de entrada, su nombre deberá ser <tabla>.odl. El procedimiento CREA.COM recibe como parámetro el nombre de la tabla. El contenido del archivo <tabla>.sql se muestra a continuación:

```
CREATE SPACE DEFINITION <tabla>
  DATAPAGES (INITIAL <Bd>,
            INCREMENT <Bdc>,
            MAXEXTENTS 9999,
            PCTFREE <100 * F> )
  INDEXPAGES (INITIAL <Bi>,
            INCREMENT <Bic>,
            MAXEXTENTS 9999)

CREATE TABLE <tabla> (
  <coll> <especificación> NULL | NOT NULL,
  .
  <coln> <especificación> NULL | NOT NULL
) SPACE <tabla>;

CREATE UNIQUE INDEX <tabla>
  ON <tabla> (coll,col2...)
  NOCOMPRESS
  PCTFREE = <100 * F> ;

CREATE PUBLIC SYNONYM <tabla>
  FOR abd.<tabla>;
```

El comando CREATE SPACE define un espacio para la tabla que será cargada. Tiene la sintaxis especificada en la sección 4.1.4. Por convención el nombre del espacio es el mismo de la tabla. El único caso en que no se genera espacio a una tabla es cuando ésta es muy pequeña pues basta asociarla a una definición llamada MINIMO, donde previamente deben definirse los requerimientos mínimos de una tabla.

Para crear la tabla se usa el comando CREATE TABLE. Los nombres de las columnas se indican en la definición del esquema, así como su especificación en cuanto a tipo, longitud y si permite nulos o no. Mediante la cláusula SPACE se asocia la tabla con la definición de espacio previamente creada.

Los índices que se determinó crear inicialmente a una tabla son los correspondientes a sus llaves primarias. La creación de índices se realiza con el comando CREATE INDEX. Por convención se nombró al índice primario con el mismo nombre de la tabla. La columna o columnas sobre las cuales se forma el índice son las referentes a la llave primaria. Se optó por generar los índices no comprimidos para disminuir el tiempo de acceso a éstos.

La creación de sinónimos se realiza con el fin de que otras claves puedan tener acceso a la tabla, sin tener que indicar el nombre de la clave que la creó. Esto es independiente de los permisos que el usuario debe tener para su acceso. Los sinónimos se generan mediante el comando CREATE SYNONYM. La cláusula PUBLIC se incluye para que el sinónimo tenga alcance global. El nombre completo de la tabla es abd.<tabla> porque la clave ABD es el creador.

Para cargar los datos de la tabla es necesario crear un archivo con el nombre <tabla>.ctl. Este archivo tiene los comandos necesarios para que la utilería ODL tome los datos del archivo de entrada y los almacene en la tabla recién creada. Este archivo debe contener los siguientes comandos:

```
DEFINE RECORD rec_<tabla> AS
    f<col1> ( char ( <long> ),
    f<col2> ( char ( <long> ),
    .
    .
    .
    f<coln> ( char ( <long> );
DEFINE SOURCE FILE
FROM base$dat:<tabla>.dat
LENGTH <long_total>
CONTAINING rec_<tabla>;

FOR EACH RECORD
INSERT INTO <tabla>
    ( <col1>,
      <col2>,
      .
      .
      .
      <coln> )
VALUES
    ( f<col1>,
      f<col2>,
      .
      .
      .
      f<coln> )
NEXT RECORD
```

La cláusula DEFINE RECORD sirve para definir el registro del archivo de datos. Por convención el nombre de este registro se forma por el nombre de la tabla y el prefijo rec_. Cada uno de los campos de este registro tiene el mismo nombre que la columna correspondiente de la tabla que formará más el prefijo

f. Todos los campos de este registro son de tipo char, independientemente del tipo de la columna, pues el archivo es de caracteres ASCII. La longitud del campo incluye el punto decimal.

La cláusula DEFINE SOURCE sirve para definir el archivo de entrada. A su nombre se le agrega la ruta lógica base\$dat donde se encuentra. La longitud del registro es la suma de todas las longitudes de sus campos.

La cláusula FOR EACH RECORD sirve para insertar un renglón de la tabla por cada registro del archivo de entrada.

Una deficiencia de la utilería ODI es que no permite cargar directamente datos de tipo DATE, por lo que si este es el caso, se debe hacer el procedimiento sobre una tabla auxiliar en donde el dato sea tipo char y posteriormente cargar la tabla definitiva mediante un SELECT anidado en la instrucción INSERT y usando la función to_date para la columna tipo DATE.

4.6.5 Administración

Como se comentó previamente, todas las tablas pertenecen a la clave ABD. Será conveniente crear una clave de ORACLE por cada sistema; estas claves podrán tener el privilegio RESOURCE, con la idea de que puedan generar alguna tabla auxiliar que agilice sus procesos, pero de ninguna manera estas tablas podrán ser entrada de algún otro proceso; es importante para la integridad de la información que el ciclo de vida de estas tablas auxiliares inicie con la ejecución del proceso y termine con el mismo.

Se debe hacer notar la importancia que tiene el administrador de la base de datos en la futura operación de ésta, pues entre sus principales funciones están dar de alta nuevos usuarios, así como vigilar la seguridad, integridad, redundancia y consistencia de la información.

Aplicación

En este capítulo se hablará del desarrollo del Sistema de Atención Personal en el SMOB ORACLE. Se describirán las etapas de análisis, diseño e implantación de este sistema así como el porqué fue desarrollada esta aplicación.

5.1 El porqué del Sistema de Atención Personal

De los muchos sistemas que existen en el CECAFI, se eligió desarrollar el Sistema de Atención Personal, que como se explicará posteriormente es una parte del Sistema de Inscripciones de la facultad de Ingeniería. Una razón para esto fue, que aunque el sistema anterior había funcionado adecuadamente durante varios años, presentaba algunas deficiencias que era posible mejorar, teniendo presente además que se contaba con amplia experiencia en su operación y mantenimiento.

Otra justificación de peso para desarrollar este sistema, fue que manejaba la información modular de la Facultad de Ingeniería, es decir, la referente a alumnos, asignaturas, grupos y profesores, entre otra. En total se utilizaron quince tablas, lo cual sirvió para probar la funcionalidad del modelo y para hacer algunos ajustes cuando fue necesario.

Finalmente, se debe mencionar que este es un sistema en línea, que es utilizado concurrentemente por funcionarios de la Facultad que esperan una respuesta casi inmediata. Estas características lo hacen un ideal para probar tanto al modelo, como la implantación y al manejador mismo, pues se requiere un favorable comportamiento entre dos puntos críticos: manejo de concurrencia y tiempo de respuesta.

5.2 Ubicación del Sistema de Atención Personal

El Sistema de Atención Personal forma parte del proceso de reinscripciones de la Facultad de Ingeniería. A través de este sistema los alumnos se reinscriben en esta Facultad semestre a semestre. Actualmente el sistema de reinscripciones está dividido en cuatro etapas o fechas las cuales se mencionan a continuación.

Primera etapa. En la primera etapa del sistema de reinscripciones se genera la información básica de los alumnos y asignaturas sobre la cual va a trabajar este sistema. La información que se genera es referente a: dobles inscripciones de los alumnos en las asignaturas, número de sorteo por medio del cual se atiende la solicitud de reinscripción de un alumno, asignaturas y antecedentes que se requieren para cursar una asignatura.

Segunda etapa. En esta etapa se efectúa una simulación de asignación de materias y la asignación de real. En la simulación se obtienen estadísticas de la demanda de los grupos y éstas son enviadas a las diferentes divisiones de la Facultad.

Las divisiones efectúan cambios de acuerdo a la demanda de los grupos. En la asignación se toman las modificaciones realizadas por las divisiones, se otorgan los grupos solicitados, se generan las primeras listas de asistencia y las formas de altas, bajas y cambios para los alumnos.

Tercera etapa. En la tercera etapa se efectúan movimientos de altas, bajas y cambios de alumnos que quedaron inconformes con su inscripción. Se genera el comprobante de inscripción de los alumnos, la situación final de las asignaturas y la situación final de los alumnos.

Cuarta etapa. Esta conforma la etapa final del proceso de inscripciones. Es donde se ejecuta el Sistema de Atención Personal. Aquí mismo se generan las listas definitivas de asistencia y la situación final tanto de alumnos como de asignaturas. También en esta etapa se generan los archivos de reinscripción requeridos por la Dirección General para la Administración Escolar.

5.3 Descripción y alcance del Sistema

Atención Personal es un sistema en línea por medio del cual se efectúan movimientos de altas, bajas y cambios a la situación final de los alumnos que están inconformes con los grupos proporcionados en las etapas anteriores, o alumnos que por causas ajenas a ellos no pudieron cubrir el proceso de reinscripción en la segunda o tercera etapa.

En este sistema intervienen los siguientes Organos de la Facultad de Ingeniería: Secretaría de Servicios Escolares

Aplicación

(SSE), Oficina de Servicios Escolares (OSE), Divisiones de la Facultad de Ingeniería (CICIS, DICE, DCE, DCI y DCSH) y el Centro de Cálculo (CECAFI). Cuenta con las siguientes funciones:

- Proporciona información de los grupos de teoría y laboratorios en los que está inscrito un alumno.
- Proporciona información personal del alumno, como es su nombre, carrera que cursa, número de sorteo del alumno y número de asignaturas en las que se encuentra inscrito actualmente.
- Permite inscribir al alumno en una nueva asignatura.
- Permite cambiar de grupo a un alumno en una asignatura.
- Permite dar de baja a un alumno en una asignatura.
- Proporciona información actual de las asignaturas, como son: nombre, grupos asociados a ésta, cupo de cada grupo, vacantes de los grupos, profesores, horarios y salones donde se imparte esta asignatura.

Algunas funciones se encuentran restringidas de acuerdo al Órgano que ejecute este sistema. Así, a las Divisiones de la Facultad sólo se les permite efectuar altas y cambios de asignaturas que pertenecen a esa División, la OSE sólo puede efectuar bajas de asignaturas y tanto la SSE como el CECAFI pueden ejecutar todas las funciones mencionadas anteriormente sin restricción.

Una descripción detallada de procedimientos, operación y uso del sistema se encuentra en los manuales respectivos.

5.4 Etapa de Análisis

En esta etapa fueron investigadas y definidas las características y los requerimientos del sistema. Los factores considerados fueron los siguientes: entradas del sistema, salidas del sistema y reglas o restricciones que lo rigen.

5.4.1 Entradas del sistema

Las entradas del sistema están constituidas por un conjunto de tablas de la base de datos y por tablas exclusivas que deben ser borradas al finalizar el sistema. Las tablas de la base de datos se enlistan a continuación y sus definiciones aparecen en las sección de Entidades y Asociaciones (§§ 2.5.1 y 2.5.2).

ACADEMICO	ADEUDA	ALUMNO
ALUMNO_EXTERNO	ALUMNO_INTERNO	ALUMNO_NUEVO_INGRESO
ASIGNATURA	CARRERA	CLASE
COMPONE	ESTUDIA	GRUPO
INSCRITO	LUGAR_DE_ADEUDO	ORGANO

Las tablas exclusivas del sistema y sus definiciones son:

- BITACORAS. Registro de todos los movimientos que se han efectuado dentro del Sistema de Atención Personal.
- DOBLES. Asignaturas en las que un alumno se ha inscrito dos veces y no las ha aprobado.
- PARALELOS. Conjunto de grupos para una asignatura que tienen el mismo horario de clase.

5.4.2 Reglas del sistema

Estas reglas o restricciones se dividen en tres de acuerdo al elemento que afectan. Estos elementos pueden ser: alumnos, asignaturas y los Organos de la F.I. que intervienen en el proceso (SSB, OSE, DICIG, OIME, OCB, OCY, OCSH).

Alumnos. En cuanto a los alumnos se refiere se tienen las siguientes reglas o restricciones:

- Se permiten movimientos de aquellos alumnos que no hayan sobrepasado el limite de tiempo establecido por el articulo 17 del Reglamento General de Inscripciones de la U.N.A.M.
- No se otorga la inscripción a una asignatura donde el alumno tenga doble inscripción.
- Se permite la inscripción de alumnos externos, esto es, aquellos alumnos que no pertenecen a esta facultad y no violan los puntos estipulados por el articulo 19 del Reglamento General de Estudios Técnicos y Profesionales de la U.N.A.M.
- El máximo número de asignaturas en las que se pueda inscribir un alumno es de siete, de acuerdo a lo establecido por la Facultad de Ingeniería.
- El sistema debe informar si el alumno tiene adeudos en algún Organó de la facultad, quedando a criterio del usuario si se permiten o no movimientos sobre este alumno.

Asignaturas. Para las asignaturas se consideran los siguientes puntos:

- Se puede asignar lugar en un grupo aunque se haya excedido el cupo de éste, quedando a criterio del usuario.

-
- ¶ Sólo se permiten movimientos de asignaturas y no de laboratorios.

Organos. Las restricciones se aplican de acuerdo al Organo que este ejecutando el sistema y son los siguientes:

Secretaría de Servicios Escolares y CECAFI.

- ¶ Puede efectuar altas, bajas y cambios de asignaturas.
- ¶ Puede efectuar movimientos de asignaturas que pertenezcan a cualquier División.
- ¶ Puede efectuar movimientos de asignaturas que no pertenezcan a la carrera del alumno.

Oficina de Servicios Escolares.

- ¶ Sólo puede efectuar bajas de asignaturas de cualquier División y de cualquier carrera.

Divisiones de la Facultad de Ingeniería (DICTG, DIMS, DCB, DCT, DCSH).

- ¶ Pueden efectuar altas y cambios de asignaturas que pertenezcan a su División. No pueden efectuar bajas de asignaturas.
- ¶ Solo se permite que realicen movimientos de asignaturas que pertenezcan a la(s) carrera(s) del alumno.

5.4.3 Salidas del sistema

El sistema requiere de las siguientes salidas:

- Situación final actualizada de los alumnos. Tabla de INSCRITO.
- Situación final actualizada de asignaturas. Tabla de GRUPO.
- Movimientos efectuados en el Sistema Atención Personal con fecha, hora y usuario que realizó este movimiento. Tabla de BITACORAS.
- Reportes de los movimientos efectuados en el Sistema Atención Personal. Básicamente son dos: reporte general de movimientos y reporte de movimientos efectuados por cada una de las Divisiones.

5.5 Diseño y Desarrollo del Sistema

Una vez realizado el análisis del Sistema de Atención Personal se concluyó que éste requiere dos tipos de información. El primer tipo de información está relacionada con el alumno, como es su número de cuenta, nombre, carrera(s) que cursa, número de sorteo, número de teorías, número de laboratorios y las asignaturas con los laboratorios que le fueron otorgadas en las etapas anteriores de reinscripción. El segundo tipo de información se relaciona con las asignaturas, como es: nombre de la asignatura, grupos que ofrece, cupo del grupo, número de vacantes, nombre del profesor que imparte la asignatura para ese grupo, horario de clase, días y salón donde se imparte esta asignatura.

Con esta información se diseñaron dos pantallas de entrada/salida las cuales se muestran en la siguientes figuras.

ATENCIÓN PERSONAL					CECAFI
Cuenta	Nombre	Curr	Series	Tpo Lab	
Mat. Gpo. Horario UNADVS			Lab. Gpo. horario UNADVS		

Bloque de ALUMNO

Bloque de INSCRITO

Figura 5.1 Pantalla de alumnos.

ATENCIÓN PERSONAL					CECAFI
Gp. Paralelos	Gp. Vc	Nombre Profesor	Horario UNADVS Sa		

Bloque de ASIGNATURA

Bloque de GRFO

Figura 5.2 Pantalla de asignaturas.

Estas pantallas fueron creadas con el generador de aplicaciones de ORACLE SQL*FORMS. La filosofía de este sistema es proporcionar al usuario un manejo natural del mismo, pudiendo actualizar la información del alumno tal como se haría en el registro de inscripción. Es decir, se puede mover al final de

las asignaturas para darle al alumno una asignatura de alta, posicionarse en cualquiera de estas para darle de baja o modificar el grupo. Gracias a las características de este generador se creó inicialmente un prototipo del sistema. En este se conjugaban las pantallas mencionadas y una programación inicial.

Posteriormente se fue modificando de acuerdo a las restricciones establecidas en el análisis y las necesidades del usuario final hasta llegar a la última versión.

También en el desarrollo de esta aplicación se efectuaron diversos programas tanto en Lenguaje C como en DCL (Digital Command Language) y SQL*REPORTWRITER ya que fue necesario automatizar algunos procesos como son: generación de vistas del sistema, generación de tablas auxiliares, generación de sinónimos, generación de reportes de bitácoras y protección de información.

Finalmente se obtuvo el Sistema de Atención Personal completo el cual cubre con todos los requerimientos establecidos. Es importante mencionar que el sistema cuenta con un conjunto de teclas funcionales que proporcionan un manejo sencillo. A continuación se muestra este grupo de teclas así como una explicación breve de su función.

Nombre de la tecla	Descripción
Teclas globales	Tienen validez en las dos pantallas.
Salir del sistema	Termina la ejecución del sistema.
Cambio de pantalla	Permite moverse de una pantalla a otra.
Siguiente campo	Permite moverse al siguiente campo de la pantalla.
Ayuda	Proporciona ayuda al usuario.

Teclas Bloque Grupo	Tienen validez exclusiva en el bloque de grupo.
Movimientos	Permite cambiarse a la pantalla de alumnos efectuando una alta o cambio de grupo.
Nueva consulta	Permite consultar una asignatura.
Siguiente registro	Permite moverse al siguiente grupo de una asignatura.
Registro anterior	Permite moverse al grupo anterior de una asignatura.

Nombre de la tecla	Descripción
Teclas bloque inscrito	Tienen validez exclusiva en el bloque inscrito.
Nueva consulta	Permite consultar a un alumno.
Baja de asignatura	Da de baja la asignatura donde se encuentre posicionado el cursor.
Campo anterior	Permite moverse al campo anterior.
Siguiente registro	Permite moverse a la siguiente asignatura del alumno.
Registro anterior	Permite moverse a la asignatura anterior del alumno.

Actualmente el Sistema de Atención Personal está liberado y listo para ser utilizado. Está totalmente documentado desde SQL*FORMS a nivel forma, bloques, campos y pasos de programación. Como documentación externa se elaboraron manuales de procedimientos, operación y usuario. Algunos comentarios adicionales están incluidos en el capítulo de conclusiones.

Conclusiones y comentarios finales

Las conclusiones que se elaboraron están clasificadas de acuerdo a cada una de las etapas del proyecto y se presentan a continuación.

6.1 Comentarios sobre el modelo

- Se comprobó que forma en que la información se manejaba, definitivamente propiciaba una enorme redundancia e inconsistencia de datos, con los consecuentes problemas que eso genera.
- Muy probablemente por lo complicado de la estructura anterior, los usuarios y programadores de aplicaciones desconocían o malinterpretaban algunas características de varios elementos de su información. La nueva representación de Sistema bajo un enfoque global facilita el mejor entendimiento de los procesos y datos que se utilizan en el CECAFI, por lo que el personal nuevo se integra a ellos con mayor facilidad; esto último es de gran importancia en un lugar con un rápido movimiento y renovación de personal.

- Otro factor que mejoró mucho el entendimiento entre los responsables de diferentes procesos fue la homogeneización de la nomenclatura utilizada.
- Ahora es más fácil para un usuario el producir nueva información, puesto que ya no depende de que un tercero le entregue determinado archivo con el formato que requiere.
- Ahora cualquier usuario puede tener una visión global de todos los datos que son manejados en el Centro, por lo que se facilita la respuesta a nuevos requisitos y se fomenta la creatividad para producir nueva información.
- Aunque para los Usuarios y Jefes de Usuarios fue difícil cambiar su conceptualización de los datos como archivos, registros y llaves, las herramientas de Modelado Orientado a Objetos fueron un auxilio invaluable al crear el Modelo.
- La actividad que más consumió tiempo fue la documentación, creación de graficas y tablas, y la actualización de todo ello al avanzar el modelado; es por eso que conviene contar con una herramienta de computo con capacidad de graficación y modificación automática de la documentación que afecte un cambio en el modelo.
- Por último, a pesar de la detallada etapa de análisis, no se espera que el modelo sea ideal, por lo que seguramente será necesario hacer modificaciones y/o adiciones cuando se detecten nuevos requisitos. A la fecha de escribir estas conclusiones ya se habían requerido algunos cambios sacados a la luz por la creación de una nueva aplicación.

6.2 Comentarios sobre la selección

- La selección se hizo teniendo como candidatos exclusivamente a ORACLE y ADABAS porque ambos manejadores, además de ser muy buenos, ya existían en la facultad. Haber incluido algún otro hubiera sido inútil, ante el factor económico que ello implicaría.
- Aunque el principal objetivo fue determinar cual es el Manejador de Bases de Datos más adecuado al ambiente y necesidades del Centro, obtuvimos un beneficio colateral que consistió en hacer y dejar escrito un estudio teórico completo de ambos manejadores, que puede servir para difundir e impulsar su uso dentro del medio académico de la facultad.
- Los dos manejadores probaron su excelente calidad. La inclinación por ORACLE se basó en su poderoso lenguaje SQL, sus herramientas de 4ª Generación que incrementan notablemente la productividad, su ambiente de administración y su gran conectividad. Esta última ventaja ya puede explotarse con la reciente instalación de la red interna del Centro que involucra equipos VMS, UNIX y PC's.

6.3 Comentarios sobre la implantación

- La meta fijada durante la fase de implantación fue explotar al máximo las capacidades del manejador de bases de datos ORACLE para que el modelo diseñado funcionara de la manera más eficiente posible. Este objetivo se cubrió, y más aún, se

dejó establecido un procedimiento completo de carga de información que incluye tanto las políticas y normas a seguir como los programas que auxilian en este proceso.

- El principal problema encontrado en esta etapa, fue la limitación del equipo de cómputo y específicamente de sus medios de almacenamiento, capaces de alojar toda la base de datos, pero insuficientes para ser ocupados por ésta y por todos los archivos tradicionales que deben seguir operando durante la fase de migración. Debido a esto no fue posible cargar la tabla PRESENTO, que contiene información concerniente a Historias Académicas y sólo se crearon índices a las llaves primarias.
- Cuando fueron escritos estos comentarios ya había sido instalado un nuevo equipo de más velocidad y mayor capacidad de almacenamiento, donde ya es posible instalar la totalidad de información de la Base de Datos.
- Es conveniente resaltar que el volumen de información ocupado en la base de datos es de orden mucho menor (95.79 MB) al ocupado anteriormente (un disco de 176 MB y dos cintas con la información de Historias Académicas).
- Es probable que la operación misma de la base de datos, así como el dinamismo de la información, determinen cambios en la implantación realizada. Deberán generarse nuevos índices, quizá se opte por crearlos comprimidos o tal vez la experiencia indique que el uso de "clusters" no es adecuado. Sin embargo la flexibilidad del modelo relacional, así como las ventajas que ofrece el manejador de bases de datos ORACLE,

deben soportar estos pequeños cambios sin afectar el funcionamiento general de los sistemas.

6.4 Comentarios sobre la aplicación

- Durante el desarrollo de la aplicación, un factor que siempre se tuvo en cuenta fue el tiempo de respuesta. En cada cambio realizado al prototipo se tomaron en cuenta diferentes elementos que contribuyeron en la disminución de este factor. Sin embargo, las consideraciones anteriores no fueron suficientes para obtener el tiempo de respuesta ideal debido a que el equipo en el que fue desarrollada la aplicación es un equipo relativamente lento y de poca velocidad.
- La presentación y forma de uso durante el tiempo de ejecución, tiene grandes ventajas con respecto al sistema anterior. Primeramente se observa que la forma de utilizar el sistema es muy natural, con esto se quiere decir que el mecanismo es muy parecido a llenar manualmente el registro de reinscripción de un alumno. Moverse a través de la pantalla, consultar la información de alumnos y asignaturas y efectuar altas, bajas y cambios a través de presionar teclas, es notoriamente más sencillo. También se ve que toda la información necesaria para el sistema se presenta únicamente en dos pantallas. Esto mejora bastante al sistema anterior ya que éste presentaba una serie de pantallas con información repetida y con muchas respuestas de confirmación que se sentían redundantes.
- Este nuevo sistema proporciona para el usuario final un conjunto de mensajes de información más completo y específico

indicando adecuadamente el problema que pudiera tener algún alumno o asignatura.

■ Otra de las ventajas del nuevo sistema es que todos los procesos involucrados en él (preparación, ejecución y mantenimiento) se encuentran automatizados. Además provee una adecuada documentación, tanto interna como externa, que fácilmente permite realizar estos procesos. La documentación externa consiste de manuales de procedimientos, operador y de usuario.

■ finalmente es conveniente mencionar que el sistema puede mejorarse de acuerdo a las necesidades del usuario; por ejemplo, actualmente no se controla la inscripción a los laboratorios, pero podría implantarse fácilmente ya que la aplicación presenta todos los medios adecuados para llevarlo a cabo. Así como este ejemplo pueden surgir algunos otros solicitados por el usuario final.

Apéndice A

El Modelo de Entidades y

Asociaciones

El Modelo de Entidades y Asociaciones es un proceso descendente usado para simplificar el procedimiento de diseño de Bases de Datos en circunstancias reales, en donde puede haber un gran número de atributos por considerar y más de una relación entre los atributos.

A.1 Conceptos básicos

El Modelo de Entidades y Asociaciones está basado en la percepción de un mundo real que se compone de un conjunto de objetos básicos llamados **Entidades**, y de **Asociaciones** entre esos objetos. El siguiente párrafo intenta clarificar esta idea.

Cuando una Empresa maneja **PRODUCTOS** que son **ENTREGADOS** por un **DISTRIBUIDOR**, cada uno de esos productores o distribuidores es una **Entidad**, y la entrega de un **PRODUCTO** por un **DISTRIBUIDOR** representa una **Asociación**. El conjunto de todos los productos que maneja la empresa es entonces un "conjunto de Entidades", y todas las Asociaciones forman un "conjunto de Asociaciones". Sin embargo, por simplicidad, se denomina Entidad a la repre-

contación del conjunto de Entidades y Asociación a la representación del conjunto de Asociaciones.

El procedimiento de modelado es el siguiente:

1. Seleccionar las Entidades (como cliente, parte, o proveedor) y las Asociaciones entre ellas (como orden de partes por clientes, partes proporcionadas por el proveedor, etc.) que están dentro del Alcance de Integración de la Empresa.
2. Asignar atributos a esas entidades y Asociaciones, de tal forma que se puedan obtener tablas bien normalizadas.

Hasta aquí han entrado en juego principalmente tres conceptos: Entidad, Asociación y Atributo. A continuación se intentan definir.

Entidad. Es un objeto que existe y que es distinguible de otros objetos. Es algo (persona, lugar, objeto, concepto) a lo que la Empresa le reconoce poder existir en forma independiente y que puede ser definido en forma única. Por ejemplo: cliente, máquina, contrato.

Atributo. Es una propiedad de una Entidad. Número, nombre y RFC pueden ser atributos de la Entidad cliente.

Asociación. Es el vínculo que existe entre dos o más Entidades. Por ejemplo, la Entidad departamento puede asociarse con la Entidad empleado via la asociación emplea. Normalmente una Empresa utiliza sujetos para describir Entidades y verbos para describir Asociaciones.

DISTRIBUIDOR VENDE PRODUCTO

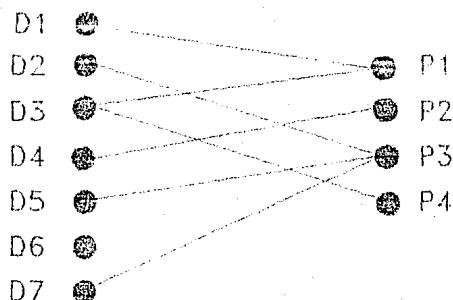


Figura A.1 Diagrama de ocurrencias DISTRIBUIDOR-VENDE-PRODUCTO.

Por definición, toda ocurrencia de una Entidad debe ser identificable en forma única. Es por lo anterior que se debe encontrar un identificador de la Entidad o llave. Existen varios tipos de llaves:

Súper llave. Es un conjunto de uno o más atributos que, tomados en conjunto, permiten identificar en forma única una Entidad dentro del conjunto de Entidades.

Llave candidato. El concepto de Súper llave no es suficiente, puesto que puede contener atributos extraños. Las Súper llaves con el menor conjunto de atributos se conocen como llaves candidato y es posible que existan diferentes conjuntos de atributos que puedan servir como llaves candidato.

Llave primaria. Una llave primaria es una llave candidato que ha sido seleccionada por el diseñador de la base de datos como

el medio de identificar Entidades. Durante el proceso de modelado se puede hablar, mas bien, de llaves candidato, puesto que es posible que en algún momento del proceso se decida seleccionar otra llave como primaria.

A.2 Diagramas de Entidades y Asociaciones

El modelado de Entidades y Asociaciones cuenta con una herramienta grafica para cumplir sus objetivos. El proceso se realiza dibujando diagramas conocidos como Diagramas de Entidades y Asociaciones (DEA). Las convenciones al dibujar DEA son:

1. Las Entidades serán representadas por rectángulos.
2. Las Asociaciones serán rombos.
3. Las líneas de conexión mostrarán qué Entidades son vinculadas por cuál Asociación.
4. Los atributos de las Entidades y las Asociaciones se muestran como círculos o elipses conectados al rombo o rectángulo correspondiente.
5. El grado de la Asociación será representado por 1, M ó N, sobre las líneas de conexión.
6. El grado de pertenencia se indicará terminando la correspondiente línea de conexión dentro de un pequeño rectángulo que forma parte de la Entidad.

Tanto los grados de asociación como las clases de pertenencia serán de utilidad al llevar el modelo de datos a un esquema conceptual. La información que proporciona define cómo estarán organizados los datos en la Base de Datos.

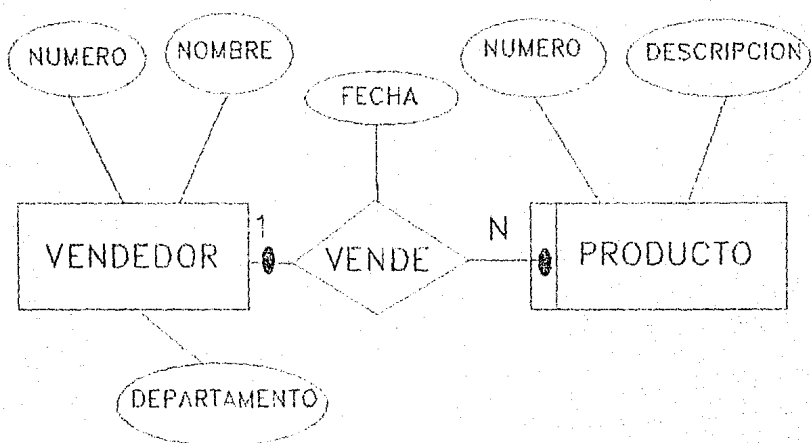


Figura A.2 Ejemplo de un DEA.

Grado de una asociación. Representa, en forma cualitativa, el número de ocurrencias de una entidad con las que puede estar asociada una ocurrencia de otra entidad. Una asociación puede tener tres tipos de grados:

TIPO		PUEDA INCLUIR
1:1	(uno a uno)	1:0 0:1 1:1
1:N	(uno a muchos)	1:0 0:1 1:1 1:N
M:N	(muchos a muchos)	1:0 0:1 1:1 1:N N:1 M:N

En un DEA, los grados de asociación se indican junto al rectángulo que representa cada Entidad, escribiendo un 1 una N o una M, según sea el caso correspondiente.

El grado de asociación puede ser obtenido de las reglas de empresa. Las reglas de empresa son definiciones que se obtienen del análisis de datos. Por ejemplo:

"Un conferencista puede dictar muchos cursos"

"Un curso sólo puede ser dictado por un conferencista"

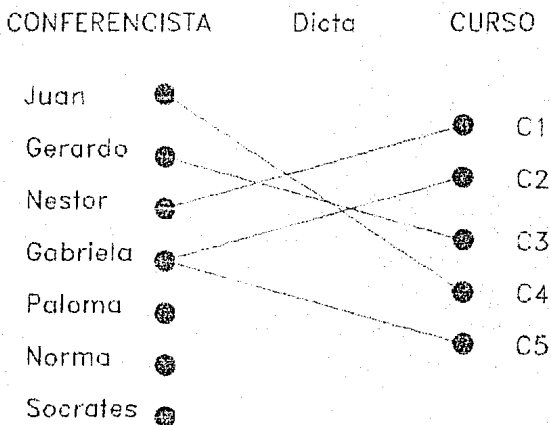


Figura A.3 Un conferencista puede dictar muchos cursos. Un curso sólo puede ser dictado por un conferencista.

Los grados de asociación definen la capacidad de determinación de una entidad hacia otra, de tal forma que en una asociación 1:1 existe determinación en ambos sentidos, en una asociación 1:N sólo existe determinación de la entidad de grado N hacia la de grado 1, y en asociaciones M:N no existe determinación.

Los grados que incluye una categoría de asociación pueden ser reducidos con reglas de empresa más estrictas, como:

"Un conferencista debe ofrecer varios cursos"

"Un curso debe ser ofrecido por un conferencista"

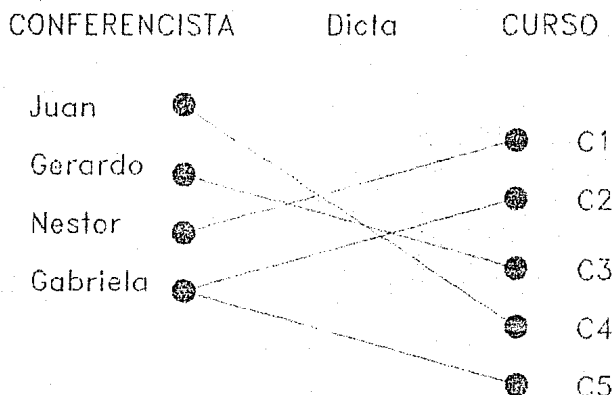


Figura A.4 Un conferencista debe ofrecer varios cursos. Un curso debe ser ofrecido por un conferencista.

Clases de pertenencia. Las reglas de empresa que excluyen Asociaciones 1:0 se traducen como la obligatoriedad de que una entidad participe en una asociación con otra Entidad. Una Entidad con obligatoriedad se puede considerar como una Entidad débil puesto que para existir depende de la existencia de otra Entidad.

La clase de pertenencia se indica con un punto en la línea de conexión, que está dentro del rectángulo que representa la Entidad, cuando ésta tiene obligatoriedad; o fuera de él, cuando ésta no la tiene.

Asociaciones multitudinarias. Conocidas como asociaciones N-arias, son asociaciones que en un momento existen entre más de dos Entidades. El Modelo de Entidades y Asociaciones representa estos casos relacionando las Entidades involucradas con un mismo rombo, que representa la Asociación.

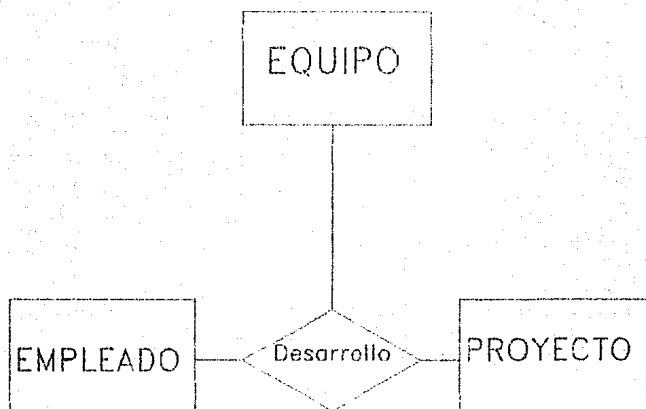


Figura A.5 Asociación ternaria PARTE-PROVEDOR-ALMACEN.

En las asociaciones multitudinarias no es de mucha utilidad especificar el grado de asociación y las clases de pertenencia; al final, la forma de organizar los datos es la misma. Sin embargo, algunas veces se incluye esa información como parte de la documentación gráfica que el DEA representa para el sistema.

Cuando la asociación se da entre una gran cantidad de Entidades, para evitar la pérdida de normalización algunos autores sugieren transformar la Asociación en una nueva Entidad agregada, creando nuevas asociaciones entre la Asociación transformada y las Entidades.

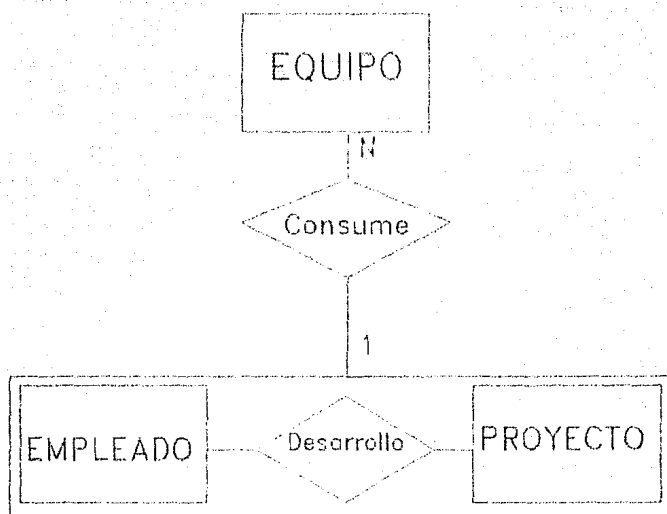


Figura A.6 Asociación multitudinaria trasformada.

Asociaciones recursivas. Son las que se dan entre Entidades del mismo conjunto de Entidades. Un ejemplo lo podemos ver considerando al conjunto de Entidades CIUDADANO y la Asociación MATRIMONIO, que debe ser entre dos CIUDADANOS. En estos casos el modelo se realiza exactamente igual a los casos no-recursivos, pero se debe indicar el papel que juega cada Entidad en la Asociación.

Generalización, especialización y agregación. Son herramientas de diseño utilizadas para facilitar la representación de la realidad y facilitar su manipulación. Por ejemplo, algunas veces será apropiado hablar de una PERSONA sin mencionar sus características, mientras que en otras se requerirá verla con más detalle, incluyendo su nombre o fecha de nacimiento.

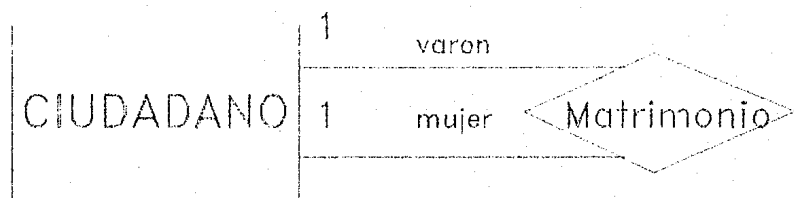


Figura A.7 Asociación recursiva CIUDADANO, MATRIMONIO.

La agregación da al diseñador la habilidad de descomponer objetos gradualmente en sus componentes detalladas, o de agregarlos en objetos de más alto nivel. Es un hecho que cuando se estudian los resultados que arroja el análisis de datos de la Empresa, se agregan datos aislados para obtener Entidades.

La **generalización** tiene que ver con la asignación de roles. Permite que los objetos sean separados en varios objetos **especializados**. Por ejemplo, un CAUSANTE pueda ser especializado en CAUSANTE MAYOR y CAUSANTE MENOR para resaltar las diferencias que existen entre ambos tipos de causante. Inversamente, MARIDO y ESPOSA pueden ser generalizados como un CIUDADANO para marcar la igualdad de atributos que ambos tienen.

En un DEA la generalización y la especialización se indican con un triángulo que inscribe a la frase "ES UN", pudiéndose leer, por ejemplo, CAUSANTE ES UN CAUSANTE MENOR y un CAUSANTE MAYOR. La diferencia entre una y otra abstracción se indica con el grosor de las líneas que unen al triángulo con las Entidades

involucradas, líneas gruesas representarían generalización, mientras que las delgadas indican especialización.

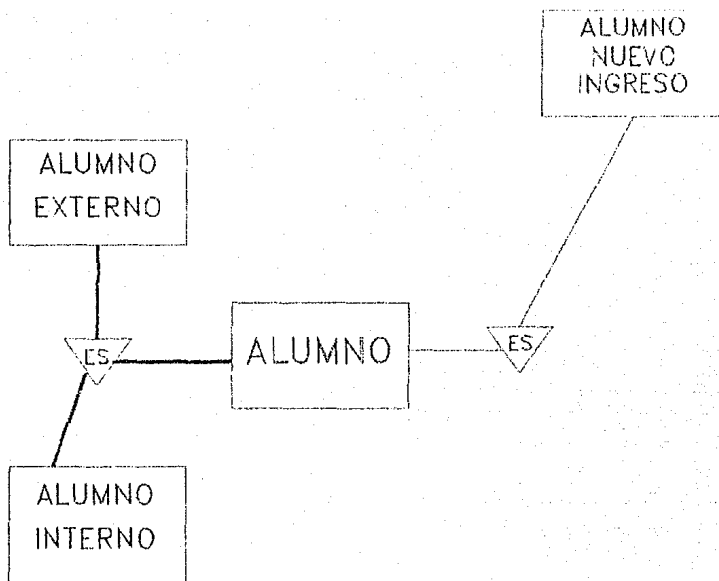


Figura A.8 Ejemplos de generalización y especialización. La línea gruesa indica generalización.

A.3 Mapeo de DEA a esqueletos y tablas

Como se anotó en la sección anterior, los atributos de Entidades y Asociaciones se muestran como elipses conectadas a los rombos o rectángulos. Sin embargo, en un DEA en donde existan muchas Entidades y Asociaciones, el número de círculos y elipses puede ser tan aparatoso que lo haría difícil de dibujar e incluso de comprender.

Para evitar el uso de elipses es generalmente conveniente representar a los atributos en tablas que se anexas al diagrama, adelantando incluso el camino hacia la implantación dentro de un ambiente Relacional.

La primera etapa de mapeo es la obtención de esqueletos de las tablas que componen el sistema. El esqueleto de una tabla se compone de: el nombre de la tabla, que usualmente es el nombre de la Entidad o Asociación; una lista de atributos mínimos que debe contener esa tabla, que por lo regular son una llave candidato y las llaves foráneas necesarias para mantener el vínculo con otras tablas; y grupos de tres puntos, que indican la futura presencia de otros atributos de la tabla. La llave candidato se coloca al principio de la lista y se subraya para indicar su calidad de identificador de la Entidad.

La segunda etapa es la asignación del resto de atributos, colocándolos en la tabla que les corresponde, y cumpliendo siempre con las reglas de normalización (3B). Es un hecho que el conjunto de tablas resultantes puede ser implantado directamente dentro del ambiente de un manejador de bases de datos relacionales, puesto que cada tabla será una relación bien normalizada.

Obtención de esqueletos. Atendiendo a cada Asociación, y observando los correspondientes grados de asociación y clases de pertenencia, se pueden obtener directamente los esqueletos de las tablas que corresponden al diagrama. A continuación se muestra esto último con algunos ejemplos.

Representación de Asociaciones 1:1

Obligatoriedad para ambas Entidades:



EMPLEADO(#empleado, ... ,#auto, ...)

#empleado y #auto son los identificadores de cada Entidad.

Obligatoriedad sólo para una Entidad:



EMPLEADO(#empleado, ...)

AUTO(#auto, ... ,#empleado)

No obligatoriedad para las dos entidades:



Se crea una tabla para la Asociación.

EMPLEADO(#empleado, ...)

AUTO(#auto, ...)

Usa(#empleado,#auto, ...)

En este caso, la llave candidato de la tabla de la Asociación puede ser la llave candidato de cualquiera de las Entidades.

Representación de Asociaciones 1:N

Obligatoriedad en la Entidad de grado N:



LUGAR(nombre, ...)

ARTICULO(#artículo, ..., nombre)

Nótese que la llave candidato de la Entidad de grado 1 se convierte en llave foránea de la tabla que representa la Entidad de grado N, en ese orden. De otra forma la llave de ARTICULO podría tomar valores nulos.

Sin obligatoriedad en la Entidad de grado 1:



LUGAR(nombre, ...)

ARTICULO(#articulo, ...)

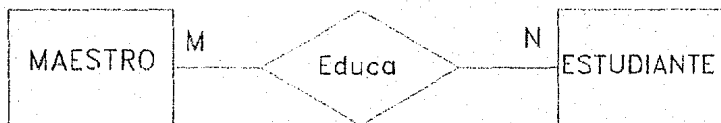
Contiene(#articulo, nombre, ...)

La llave candidato de la Entidad de grado 1 se convierte en la llave foránea de la Tabla que representa a la Asociación, en ese orden.

Los casos en que existe o no obligatoriedad en la Entidad de grado 1 no modifican nada.

Representación de Asociaciones M:N

No importa la existencia o no de obligatoriedad en las Entidades:



MAESTRO(nombre, ...)

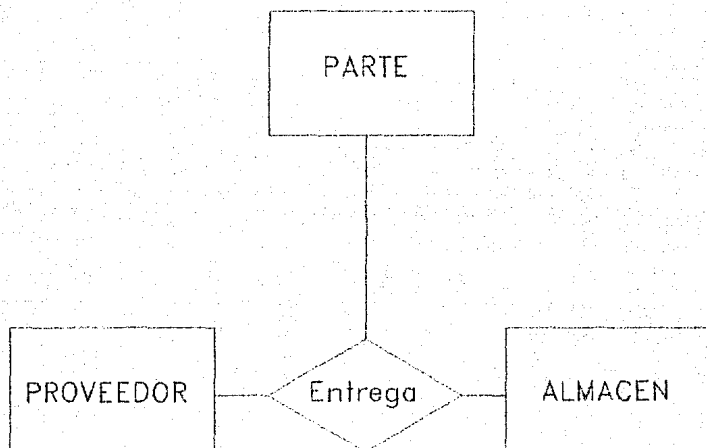
ESTUDIANTE(#estudiante, ...)

Educa(nombre, #estudiante, ...)

La tabla que representa la Asociación debe tener como llave candidato la concatenación de las llaves de las Entidades.

Representación de Asociaciones Multitudinarias

No importan mucho los grados de asociación o de pertenencia, de cualquier forma se debe representar la Asociación y formar una llave primaria con la concatenación de los identificadores de las Entidades que participan.



```

PARTE(#parte, ... )
ALMACEN(#almacén, ...)
PROVEEDOR(#proveedor, ...)
Entrega(#proveedor,#parte,#almacén, ...)
    
```


Descomposición de Asociaciones M:N. Para algunos modelos o manejadores de bases de datos existe la imposibilidad de representar las Asociaciones M:N, debido por ejemplo a la incapacidad de concatenar llaves. En estos casos es posible descomponer una Asociación M:N en dos 1:N, lo cual no significa que dos Asociaciones 1:N deban representar una M:N. Ejemplo:

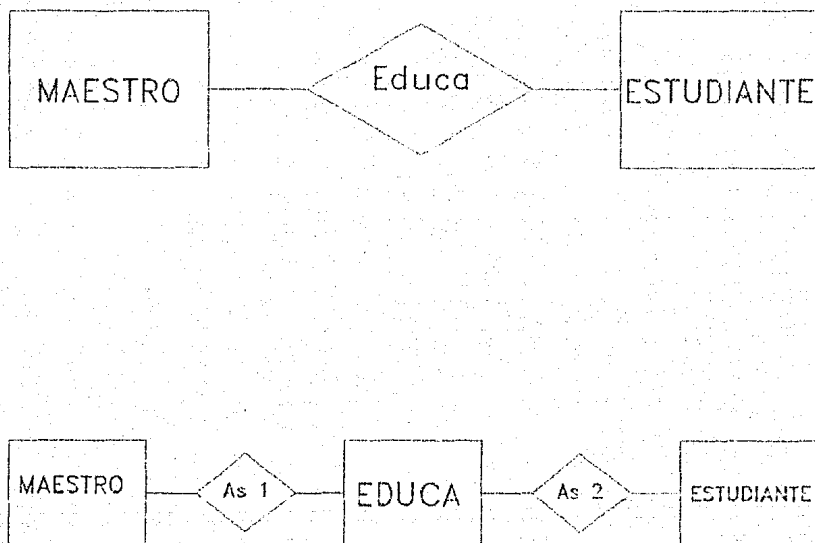


Figura A.16 Descomposición de una Asociación M:N en dos 1:N.

Apéndice B

Normalización

Normalización es una técnica desarrollada para asegurar que las estructuras de datos sean eficientes. Los beneficios de la Normalización son:

- 1 Libera de dependencias indeseables de inserción, borrado y actualización.
- 2 Minimiza la reestructuración de datos cuando se introduce algo nuevo. Se mejora la independencia de datos, permitiendo que las extensiones a la base de datos tengan poco o ningún efecto sobre los programas o aplicaciones que tienen acceso a ella.
- 3 No se introducen restricciones artificiales a las estructuras de datos.

Aunque se han definido más estados de normalización, sólo se han aceptado ampliamente tres. Estos se conocen como primera, segunda, y tercera formas normales, o 1NF, 2NF y 3NF respectivamente.

El normalizar es un proceso ascendente, en el que se parte de un universo de relaciones y atributos, y se avanza de forma en forma hasta llegar a la tercera forma normal.

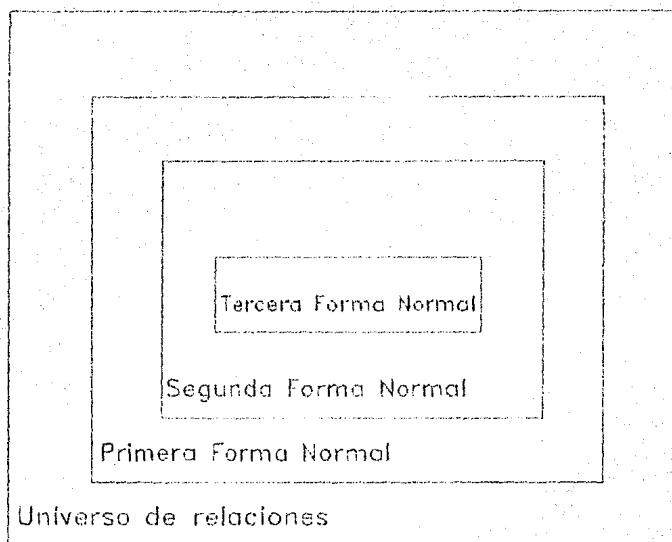


Figura B.1 El universo de las relaciones.

Las etapas de normalización se muestran adelante con un ejemplo, dada la relación no normalizada:

```
ORDEN ( #orden, fecha, #proveedor, nombre_proveedor, direc-
        ción_proveedor, #producto, descripción_producto,
        cantidad_producto, precio_total_producto,
        precio_total_orden )
```

Primera forma normal. Un registro en primera forma normal no incluye grupos repetidos.

En la relación ORDEN, se observa que para una misma orden habrá varios productos, por lo que #producto y otros atributos serán grupos repetidos. En 1NF habría que separar:

```
ORDEN ( #orden, fecha, #proveedor, nombre_proveedor,
        dirección_proveedor, precio_total_orden )
PRODUCTO_ORDENADO (
    #orden, #producto, descripción_producto, precio_
    producto, cantidad_producto, precio_total_producto
)
```

Segunda forma normal. Cada atributo depende de la totalidad de la llave, y no de sólo una parte de ella.

Se puede observar en PRODUCTO_ORDENADO que descripción_producto depende sólo de #producto, y no tiene que ver con #orden. En 2NF quedaría:

```
ORDEN ( #orden, fecha, #proveedor, nombre_proveedor,
        dirección_proveedor, precio_total_orden )

PRODUCTO (
    #producto, descripción_producto, precio_producto
)
PRODUCTO_ORDENADO (
    #orden, #producto, cantidad_producto,
    precio_total_producto )
```

Tercera forma normal. Cada atributo no primo (que no es parte de la llave) no tiene dependencia transitoria sobre la llave.

En la relación ORDEN se presenta este problema:

```
ORDEN ( #orden,
        fecha,
        #proveedor,
        nombre_proveedor,
        dirección_proveedor,
        precio_total_orden
    )
```

De modo que las tablas en 3NF quedarían como:

```
ORDEN ( #orden, fecha, #proveedor, precio_total_orden )
PROVEEDOR (
    #proveedor, nombre_proveedor, dirección_proveedor
)
PRODUCTO (
    #producto, descripción_producto, precio_producto
)
PRODUCTO_ORDENADO (
    #orden, #producto, cantidad_producto,
    precio_total_producto
)
```

Bibliografia

Cohen, Leo

Data Base Management Systems.

Date, C. J.

An Introduction to Database Systems.

Addison-Wesley, (1983)

Hawryszkiewicz, T. J.

Database Analysis and Design.

Science Research Associates, Inc., (1984)

Howe, D. R.

Data Analysis for Database Design.

Edward Arnold Pub., (1983)

IBM Data Processing Division

Data Processing Glossary.

Jackson, Glenn A.

Relational Database Design with Microcomputer Applications.

Prentice-Hall, (1988)

Bibliografia

Adlin, Henry F., SILVERSTEIN, Abraham

Database System Concepts.

McGraw-Hill, (1986)

Martin, James

Computer Data-Base Organization.

Prentice Hall, (1975)

Martin, James

Principles of Data-Base Management.

Prentice Hall, (1976)

Mayne, Alan

Introducing Relational Database.

The National Computing Centre Limited, (1983)

ORACLE Corporation

SQL*PLUS User's Guide. V.2.0

Part Number 3201-V2.0

ORACLE Corporation

The Oracle Database Administrator's Guide.

Part Number 3601-V5.1

Pressman, Roger S.

Software Engineering a Practitioner's Approach.

McGraw-Hill, (1982)

Senn, James A.

Analysis and Design of Information Systems.

McGraw-Hill, (1984)

Software AG.

ADABAS (VMS) Command Reference Manual. V.1.5

ADV-150-050

Software AG.

ADABAS (VMS) DBA Reference Manual. V.1.4

ADV-140-030

Software AG.

ADABAS (VMS) Instalation and Operations. V.1.5

ADV-150-018

Software AG.

ADABAS (VMS) Security Manual. V.1.5

ADV-150-090

Software AG.

ADABAS (VMS) Utilities. V.1.4

ADV-140-030

Software AG.

NATURAL (VMS) Reference Manual V.1.2

NAV-120-030

Bibliografia

Chen, Peter Pin-San.

Applications of the Entity-Relationship Model,
NYU symposium on Database Design, pp. 25-33.
New York University, New York City, may 1978.

Cohen, Leo

Data Base Management Systems.

Date, C. J.

An Introduction to Database Systems.
Addison-Wesley, (1983)

Hawryszkiewicz, I. T.

Database Analysis and Design.
Science Research Associates, Inc., (1984)

Howe, D. R.

Data Analysis for Database Design.
Edward Arnold Pub., (1983)

IBM Data Processing Division

Data Processing Glossary.

Jackson, Glenn A.

Relational Database Design with Microcomputer Applications.
Prentice-Hall, (1988)

Kahn, Beverly K.

A structure Logical Database Design Methodology,
NYU symposium on Database Design, pp. 15-24.
New York University, New York City, may 1978.

Korth, Henry F., Silverschatz, Abraham

Database System Concepts.
McGraw-Hill, (1986)

Martin, James

Computer Data-Base Organization.
Prentice Hall, (1975)

Martin, James

Principles of Data-Base Management.
Prentice Hall, (1976)

Mayne, Alan

Introducing Relational Database.
The National Computing Centre Limited, (1983)

Mittra, Sitansu S. Dr.

Evaluation and Selection of a DBMS for Academic Use,
Journal of Systems Management.

ORACLE Corporation
*SQL*PLUS User's Guide. V.2.0*
Part Number 3201-V2.0

ORACLE Corporation
The Oracle Database Administrator's Guide.
Part Number 3601-V5.1

Palmer, I. R.
*Practicalities in Applying a Formal
Metodology to Database Analysis,*
NYU symposium on Database Design, pp. 77-84.
New York University, New York City, may 1978.

Pressman, Roger S.
Software Engineering a Practitioner's Approach.
McGraw-Hill, (1982)

Senn, James A.
Analysis and Design of Information Systems.
McGraw-Hill, (1984)

Smith, John Miles, Smith, Diane C.P.
Principles of Database Conceptual Design.
NYU symposium on Database Design, pp. 114-146.
New York University, New York City, may 1978.

Software AG.
ADABAS (VMS) Command Reference Manual. V.1.5
ADV-150-050

Bibliografia

Software AG.

ADABAS (VMS) DBA Reference Manual. V.1.4

ADV-140-030

Software AG.

ADABAS (VMS) Instalation and Operations. V.1.5

ADV-150-018

Software AG.

ADABAS (VMS) Security Manual. V.1.5

ADV-150-090

Software AG.

ADABAS (VMS) Utilities. V.1.4

ADV-140-030

Software AG.

NATURAL (VMS) Reference Manual V.1.2

NAV-120-030