

45

UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO

ESCUELA NACIONAL DE ESTUDIOS PROFESIONALES
"ACATLAN"

ANALISIS COMPARATIVO DEL MODELO
ORIENTADO A OBJETOS PARA REDISEÑAR UNA
BASE DE DATOS RELACIONAL.

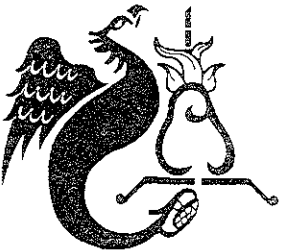
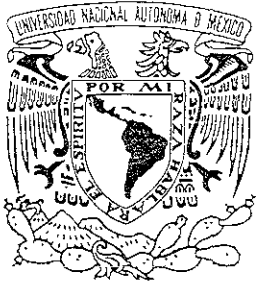
SEMINARIO - TALLER
EXTRACURRICULAR
QUE PARA OBTENER EL TITULO DE:
LICENCIADO EN MATEMATICAS
APLICADAS Y COMPUTACION
P R E S E N T A :
SERGIO ALFREDO YEVERINO GONZALEZ

ASESOR. LIC. JUAN TORRES LOVERA

ACATLAN, EDO. DE MEXICO. OCTUBRE DE 2001

ESTA TESIS NO SALE
DE LA BIBLIOTECA

DE LA ESCUELA





Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

INTRODUCCION	4
1.GENERALIDADES DE LA TECNOLOGÍA ORIENTADA A OBJETOS.	7
1 1. ANTECEDENTES	7
1 2. LA ORIENTACIÓN A OBJETOS	9
1 3. MODELO DE OBJETOS	10
1 4. PROPIEDADES GENERALES DE LA ORIENTACION A OBJETOS	15
1 5. ELEMENTOS DE ORIENTACIÓN A OBJETOS	15
1 5 1 OBJETOS	15
1 6 MÉTODOS DE ANALISIS ORIENTADOS A OBJETOS	19
1 7 CLASIFICACIÓN DE LAS BASES DE DATOS ANTE LA ORIENTACIÓN A OBJETOS	24
1 8. EXTENSIÓN DE LOS MODELOS ORIENTADOS A OBJETOS	26
CONCLUSIONES	27
2.PLANTEAMIENTO DE DIFERENTES MODELOS EN COMPARACIÓN CON EL MODELO DE DATOS ORIENTADOS A OBJETOS, EN CONJUNTO CON LAS BASE DE DATOS.	28
2 1. MODELO ENTIDAD-RELACIÓN	28
2 2 ESTUDIOS PARA IMPLEMENTAR UNA BASE DE DATOS ORIENTADA A OBJETOS	34
2 3. COMPARACIONES CON OTROS MODELOS	34
2 4 EL MODELO DE DATOS Y LAS BASES DE DATOS	37
2 5 CONCEPTOS BÁSICOS DEL MODELO DE DATOS ORIENTADO A OBJETOS	37
2 6 ¿QUÉ ES UN BASE DE DATOS ORIENTADA A OBJETOS?	40
2 7 CARACTERÍSTICAS GENERALES DE LAS BASES DE DATOS ORIENTADAS A OBJETOS	43
2 8. MODELO ODMG-3	44
CONCLUSIONES	52
3 EJEMPLO DE UNA BASE DE DATOS UTILIZANDO EL MODELO ORIENTADO A OBJETOS.	53
3 1 ESCENARIOS Y LA ORIENTACIÓN A OBJETOS	53
3 2 MODELO CENTRAL	54
3 3 APLICACION	59
3 4 COMO SELECCIONAR UN MODELO	64
3 5 LAS BASES	65
3 6 LA TECNOLOGIA DE OBJETOS Y DBMS	65
3 7. LA TECNOLOGIA DE DBMS	66
3 8 ESCOGER EL LENGUAJE	66
3 10 CARACTERISTICAS DE UN DBMS	67
3 11 LAS CARACTERISTICAS DE INTERACCION	67
3 12 DE QUE MANERA COMPRENDER LOS REQUERIMIENTOS DE LAS APLICACIONES	68
3 13 ETAPA FINAL	68

3 14 CARACTERISTICAS CRITICAS	68
3 15 CREAR UNA LISTA DE VENEDORES	69
CONCLUSIONES	70
CONCLUSIONES FINALES	71
ESCENARIOS	73
GLOSARIO	76
FUENTES DE CONSULTA	78

*Dedicado a mis padres Sergio e Irma y a mi hermano Jesús por ser mis
mejores amigos y maestros.*

INTRODUCCION

Ubicar la tecnológica orientada a objetos, puede llevarnos a descubrir diferentes ramas de estudio, dentro de las cuales se encuentran los lenguajes de programación, metodologías de análisis, diseño, herramientas, bases de datos, sistemas de cómputo.

Debido a ésta división, el autor se ha interesado en analizar las bases de datos y sus modelos, de forma más específica se compara los diferentes modelos de datos en relación a un modelo orientados a objetos para sustituir un modelo entidad-relación de una base de datos, el desarrollo a corto, mediano y largo plazo que ofrecerán las bases de datos será contar con un medio masivo para conocer lugares, países, datos personales y sobre todo pasará un individuo de ser sólo un CURP(Clave Unica de Registro Poblacional) a un DBCURP(Base de Datos de Clave Unica de Registro Poblacional) en donde por medio de la computadora y del INTERNET, se ubicará a la persona y todo lo que concierne a ella, por ejemplo: nombre completo, lugar de trabajo, residencia, lugares y comida preferidos, tipo de sangre entre una infinidad de datos; por lo tanto sin deslindar los problemas que un día puede traer consigo una base de datos masiva y que no solamente pueda ser utilizada en las personas sino en bibliotecas, libros enteros, trabajos de investigación, venta de coches, se proponen diferentes modelos para compararlos con el modelo de datos orientados a objetos, que describan sus características, aunque la complejidad es variada y difícil de asimilar, se sugiere conocer cual es la funcionalidad que proporciona cada modelo, así como identificar las características propias de la orientación a objetos, objetos, herencia, jerarquía, clases, polimorfismo, así como la metodologías más comunes, por lo tanto; el objetivo será dar a conocer los diferentes modelos, para después elegir uno de ellos, si así lo desea, y se preguntarán por que modelos, los modelos que son representaciones conceptuales, físicas o abstractas de la vida real, en este caso el modelo que se representa es un modelo de datos conceptual con características abstractas, que define los inicios o la primera fase para el desarrollo de una base de datos dentro de un sistema de

cómputo, así se observa que la relación entre bases de datos y modelos nunca debe deslindarse uno del otro, de forma descriptiva el modelo representa la forma conceptual del mundo real, y las bases de datos establecen todas las reglas para hacer funcionar el modelo y almacenar información.

Por lo tanto al leer el siguiente trabajo, no llegarán a ser expertos en bases de datos o en la creación de modelos, pero si, puedan establecer criterios para que en cierto momento puedan decidir con que modelo trabajar y bajo que base de datos.

Otra herramienta, que puede ser de mucha utilidad, y la cual no es un concepto abstracto y que sin embargo esta como un objeto más para realizar cualquier, análisis, metodología, plan de estudios, selección del modelo será la planeación, hacer énfasis en la planeación no es cosa de enumerar ciertos puntos en los cuales se cumpla con estricta rigidez cada uno de ellos, hablar de planeación hacia la selección de un modelo de datos consistirá en retomar diferentes características y compararlas con otros modelos, hacer énfasis mas en un modelo que en otro, tratar de describirlo de forma mas práctica, conformar un equipo de trabajo y sobre todo llegar a un plan fijo en donde se enumeren los casos con mayor probabilidad de éxito como los no exitosos

A lo largo de veinte años la computación ha sido desarrollada, en donde existen estudios son en las bases de datos. y éstas están relacionadas en todos los ámbitos, las encontramos en el sector privado, en el sector federal, en las universidades, en los institutos, en fin, en cualquier pequeña o mediana empresa, hasta en nuestras propias casas, su finalidad; almacenar información de manera ordenada.

Las bases de datos han sufrido cambios, en sus modelos, en su arquitectura, creación, conceptos, manejadores. En el presente trabajo se adentró en los modelos de datos, ya que son el pilar para el desarrollo físico de las bases de datos. en donde los modelos de datos, modelos conceptuales, modelos que representan de manera mas precisa el mundo real, en donde cada institución refleja los datos y la información que será primordial para la toma de decisiones

Los modelos que desde sus inicios han sido elaborados bajo fundamentos científicos, ya comprobados, como el modelo entidad-relación, no es así el caso del modelo orientado a objetos, por lo cual diferentes consorcios de programadores, diseñadores, empresarios, expertos han elaborado diferentes modelos de datos orientados a objetos que dificultan la selección de uno para su uso comercial.

Por tal motivo el autor expone una idea general de los modelos de datos, la evaluación de diferentes modelos y la aplicación de uno de ellos en una base de datos, tomando ideas del modelo entidad-relación para que en determinado momento se reemplace por el modelo orientado a objetos, siempre y cuando el modelo orientado a objetos represente ya no un paradigma para las instituciones sino todo lo contrario que sea establecido y utilizado en universidades, institutos de investigación y por último en instituciones privadas y de gobierno.

Como el desarrollo de un niño nacimos con el modelo funcional, hemos crecido con el modelo jerárquico y de red, alcanzamos una etapa de maduración con el modelo entidad-relación, se encontrará con el modelo orientado a objetos un crecimiento completo que se establezca en cada institución o se seguirá utilizando el modelo entidad-relación con algunas combinaciones de objetos que no signifique la idea purista de objetos. Para poder enriquecer estos conceptos se invita a formar parte de la difícil tarea de selección del modelo de datos orientados a objetos, así como el conocimiento de algunos de los conceptos que se muestran aquí, ya que en realidad hay muy pocas investigaciones sobre el modelo de datos orientados a objetos. Por lo tanto, que las ideas y la creatividad de una investigación, comparta con cada uno de ustedes, las maravillas de los modelos y por supuesto de la Orientación a Objetos.

1.GENERALIDADES DE LA TECNOLOGÍA ORIENTADA A OBJETOS.

1.1. ANTECEDENTES

En los años setenta y ochenta el desarrollo de sistemas se basaba en la programación estructurada, a finales de los años ochenta y principios de los noventa surge una nueva metodología llamada "orientación a objetos", pero tomemos en cuenta lo que afirman Ten Dyke y Kunz(1989)¹, que los diseñadores del misil Minuteman utilizaron técnicas rudimentarias de la orientación a objetos en el año de 1957, pese a que la orientación a objetos no ha sido utilizada de manera mas comercial por los desarrolladores si no por un grupo muy pequeño de científicos, la orientación a objetos nace con el lenguaje de programación SIMULA, con la programación de *sucesos discretos*, en Noruega, en el año de 1967.Y continua con el lenguaje desarrollado en el centro de investigación de Palo Alto,California PARC(Palo Alto Research Centre) que toma como base los objetos nombrado SMALLTALK, y no solamente toma sus orígenes de SIMULA si no que también debemos su creación de la tesis doctoral de Alan Kay al tener como visión una computadora personal, pero universal, capaz de tratar cualquier clase de problema de gestión de información y ser manejado por cualquier persona: a este lenguaje se le han sumado lenguajes como Alphard(1976) y el CLU(1977) que se han encargado de difundir la orientación a objetos. El origen mas preciso de la orientación a objetos en donde se puede desprender de conceptos mas avanzados fue la utilización del modelo de simulación el cual era un problema para los programadores de lenguajes convencionales de tercera generación, debido a que el programador requiere adaptar el flujo de control funcional normal en esos lenguajes a un flujo de control que describa un modo mas natural, en términos de objetos complejos, que cambian de estado e influyen sobre los sucesos a cada momento En la programación orientada a objetos, este flujo funcional es reemplazado por el paso de mensajes entre objetos, lo que originan cambios en el estado de objeto, por lo tanto la

¹Verse las: Graham: Metodos Orientados a Objetos Ed Addison Wesley/Diaz Santos pag 3

programación orientada a objetos es un enfoque sumamente natural, ya que la estructura de los programas refleja directamente la estructura del problema.

Además en los problemas de simulación existen objetos que van desde un avión, una casa, una fabrica, un virus, una línea de producción, lo que podemos decir que son objetos del mundo real y no objetos abstractos, que quizá no siempre sea fácil seleccionarlos para el desarrollo de programas

Para los años 80 con la creación de interfaces de usuario (US, user interface), los pioneros comerciales como Xerox, y después Apple, introdujeron el mundo de las interfaces universales WIMP², lo que muchas de estas ideas fueron tomadas de Smalltalk. Esta nueva tecnología³ se ha popularizado, por lo que muchos desarrolladores se han involucrado en la creación de programas, sistemas operativos, herramientas, bases de datos(DB)⁴, y demás software

Pero realmente los desarrolladores se han olvidado de las metodologías convencionales(estructurada) a la metodología orientada a objetos(MOO), es difícil establecer cual es el momento adecuado para actualizar la(s) metodologías que componen un sistema, que nos ayude a tener un sistema mas confiable, integro, seguro y eficaz.

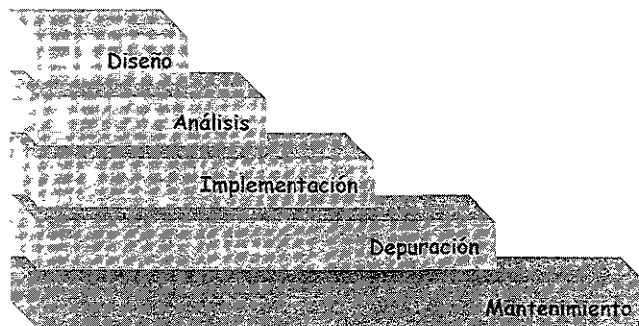


Figura 1.1 Fases de Diseño

El problema radica en la etapa de mantenimiento ya que ocupa el 67%⁵ del ciclo de vida de un proyecto de software(véase Fig.1 1)

Así que si se requiere renovar las aplicaciones dependerá del tiempo en que los sistemas lo predispongan, el hecho es que seguramente el desarrollo de un sistema nos llevará a elaborar programas, bases de datos, algún sistema operativo y la instalación de alguna red.

² WIMP viene de Windows, Icons, Mice and Pointers y se refiere al estilo de interfaz Grafico de Usuario(CUI) Véase Ian Graham "Múltiples Orientagos a Objetos" Ed Addison Wesley/Diaz Santos pag 4

³ El termino "tecnología de objetos" se usa a menudo para encerrar todos los aspectos de una visión orientada a objetos e incluye el análisis, diseño y métodos de prueba, lenguajes de programación, acorramientos, bases de datos que fueron creados usando el enfoque de sistemas Jovanka Ananda "Luis Programación Orientada a Objetos" Ed Mc Craw Hill

⁴ Por sus siglas en ingles Database

⁵ Jovanka Ananda, Luis Programación Orientada a Objetos España Ed Mc Craw-Hill, La crisis de software pags 6-8

Demasiados conceptos, metodologías, planes, capacidades humanas, técnicas, se ven involucradas en cada una de estas tecnologías de software y hardware, que implica estudios detallados especializados y de carácter analítico, técnico y descriptivo que se sería inapropiado reunir todas en un solo documento

Pero si podemos dividir las y enfocarnos a una de las partes importantes que son las bases de datos, definiéndolas de manera simple, son objetos que sirven para almacenar datos, se consideran la médula espinal de toda empresa ya que guardan dentro de su estructura toda la información que la empresa genere, independientemente de los procesos que tenga que realizar, por lo tanto una base de datos⁶ "es un conjunto, colección o depósito de datos almacenados en un soporte informático no volátil"

Las bases de datos han evolucionado y también llevan de la mano la tecnología orientada a objetos; por consiguiente que entendemos por orientación a objetos, cuales son sus elementos, cuál es el modelo de datos con el cual se desarrollan las bases de datos, cuales son sus metodologías más conocidas mencionando principalmente OMT⁷(Técnicas de Modelado de Objetos) y Coad/Yourdon así como los diferentes tipos en donde se encuentran las bases de datos frente a la tecnología OO (Orientada a Objetos).

1.2. LA ORIENTACIÓN A OBJETOS

Sin llevar ningún nombre como metodología, análisis, diseño, implementación, programas, bases de datos, herramientas, solo diremos que la Orientación a Objetos, es una nueva forma de pensar y diseñar aplicaciones que nos pueden ayudar a resolver problemas del desarrollo del software, la Orientación a Objetos⁸ puede describirse como el "conjunto de disciplinas(ingeniería) que desarrollan y modelan software que facilitan la construcción de sistemas complejos a partir de componentes"(los problemas del mundo real sean expresados de modo fácil y natural), para García Zavala Angel, Zamucio Roldán Juan Manuel entre otros definen la orientación a objetos de la siguiente manera: " Es una manera de pensar, otra manera de resolver un problema; lo más reciente en metodologías de desarrollo de software. Es un proceso mental humano aterrizado en una computadora. Antes se adecuaba el usuario al entendimiento de la computadora. Actualmente se le enseña a la computadora a entender el problema "

⁶ De Miguel Adalberto Fundamentos y modelos de las bases de datos Ed Ra-Ma pag 27

⁷ Por sus siglas en inglés Object Modeling Techniques(OMT)

⁸ Lozada Aguilar Luis Programación Orientada a Objetos Ed Mc Graw Hill pag 18

1.3. MODELO DE OBJETOS

El mundo esta formado por objetos , objetos que representan medios de transporte, transacciones bancarias, la credencial para votar que integran un determinado número de datos, los complejos sistemas de medición, en fin hay un sin número de elementos u objetos que representan los modelos.

En la actualidad necesitamos de esos modelos para representar nuestras bases de datos , la recopilación de información , ayuda a la administración y planeación para evaluar estrategias y servicios que den mejores respuestas a cada usuario.

Por ejemplo contamos con la base de datos de los bancos. la de PEMEX, la del Sistema de Ahorro para el Retiro con la creación de las AFORE, la del IMSS, la del Instituto Federal Electoral que son un reflejo de la realidad, información que forma cada uno de nosotros, información que tiene que ser estructurada para representar los modelos de datos y que por último se desarrolle en las bases de datos.

Por tal motivo es importantes especificar las bases teóricas o conceptuales de los modelo de datos para el desarrollo de cada sistema

1.3.1. FUNDAMENTOS DE LOS MODELOS DE DATOS

Básicamente constan de tres principios base⁹

1. Los datos se organizan a fin de atender las necesidades de cada proceso.
2. Responden a los requisitos de un conjunto de procesos.
3. Interpretación de la realidad con el fin de captar su semántica(significado).

1.3.2. ¿QUÉ ES UN MODELO DE DATOS?

Ian Graham" Métodos Orientados a Objetos" define el modelo ,como un formalismo matemático que consta de una notación para describir los datos y la estructura de datos(información), y de un conjunto de operaciones válidas que se pueden utilizar para manipular estos datos

Para los autores del libro "Diseño y Administración de Bases de Datos" Garry W.Hansen y James V Hansen

"Un modelo es una representación de la realidad que conserva sólo los detalles relevantes."

Ejemplos:

- Modelo Funcional (Shipman 1981).

La idea de Shipman esta muy relacionada de forma muy cercana con las ideas de las redes semánticas y la inteligencia artificial y con la programación funcional

- Modelo Jerárquico.

⁹ De Miguel Adoracion Fundamentos y modelos de Bases de Datos Ed Ra-Ma pag 80

- Modelo de Red.
- Modelo Entidad Relación
- Modelo Orientado a Objetos

Por ejemplo al momento de inscribirse en cualquier universidad, para el personal administrativo los únicos datos relevantes que les interesa podrían ser los siguientes:

- Nombre
- Apellido Paterno
- Apellido Materno
- Grupo
- Fecha
- Plantel
- Número de Cuenta

Pero existen otros datos que no se contemplan

- El tiempo en el que estuvieron formados.
- El tiempo en que se llevaron a formar su grupo
- La identificación de materias grupos y maestros
- El tiempo que han invertido para poder inscribirse
- El número de estudiantes en espera
- Las condiciones del clima

Por lo tanto el conjunto de elementos irrelevantes¹⁰ para el personal encargado de las inscripciones no son de importancia únicamente aquellos que cumplan con las características que exige su sistema de bases de datos

De otra manera los datos que para unos son relevantes para otros quizá no lo sean.

Al identificar que menú será el mas conveniente en un restaurante y un dato importante para el programador será el clima.

O quizá en otro sistema les preocupe disminuir el número de personas esperando en los cajeros, en las ventanillas, necesitan planificar el número de ventanillas o minimizar el tiempo de espera al cliente

Por lo que diferentes usuarios pueden tener diferentes modelos de la realidad

Y en donde se incorporan estos modelos en las bases de datos definida por Adoración¹¹ de Miguel y Mario Piatinni en su libro "Fundamentos de Bases de Datos" Colección de datos integrados . almacenados ,los cuales serán compartidos por diferentes usuarios y aplicaciones, formado por datos y la estructura de datos que *refleja el como están almacenados representa el modelo de datos*"

¹⁰ Hansen W Clay Diseño y Administración de Bases de Datos España Ed Prentice Hall segunda edición Realidad y modelos pags 85-87

¹¹ De Miguel Adoración Piatinni Manó Fundamentos y Modelos de Bases de Datos España Ed RA-MA pags 25-28

En un sistema SGBD (Sistema de Gestión de Bases de Datos), se encargará:
Administrar

Usuarios que a su vez puedan

 Accesar

 Registrar *y que*

 Manipular en diferentes formas

Datos que representan el modelo de la realidad

Los modelos de datos¹² serán entonces un conjunto de conceptos que permiten describir a distintos niveles de abstracción, y se dividen:

1.3.2.1. MODELOS EXTERNOS

Nos permiten representar los datos que necesita cada usuario, en particular con las estructuras propias del lenguaje de programación que va a emplear.

1.3.2.2. MODELOS GLOBALES O CONCEPTUALES.

Ayudan a describir los datos para el conjunto de usuarios , podríamos decir que es la información a nivel empresa

Los modelos globales se clasifican

1. Modelos Conceptuales

Facilitan la descripción global del conjunto de información de la empresa con independencia de la máquina (tanto de Hardware como del administrador de la base de datos), por lo que sus conceptos son muy cercanos a la realidad (entidades, atributos, interrelaciones) y se utilizan como herramientas de análisis no de implementación.

2. Modelos Convencionales

Este modelo soportado por los sistemas administradores de bases de datos, están orientados a describir los datos a nivel lógico a nivel de Sistema de Gestión de Bases de Datos, por lo cual los llamados modelos de bases de datos, son propios de cada SGBD Ejemplos

Objetos , herencia, clase del modelo Orientado a Objetos.

Tablas, atributos, relaciones, tuplas del modelo Entidad Relación

Redes en el modelo CODASYL

Arboles el modelo jerárquico

3. Modelos Internos

O modelos físicos están orientados a la máquina, siendo sus elementos de descripción punteros, índices, agrupamiento , tablas , objetos.

¹² De Miguel Adoracion Platta. Marco Fundamentos y Modelos de Bases de Datos España Ed RA-MA pag 3

El modelo de datos sea lógico o sea físico es el instrumento que se aplica a los datos para obtener el esquema.

El esquema¹³ se entiende como la descripción de la estructura de la base de datos y la realización u ocurrencia del esquema, serán los datos almacenados en el esquema en un determinado momento, el esquema no varía mientras no varíe los datos del mundo real, mientras que la ocurrencia de un esquema, es decir los datos contenidos en él son distintos en el transcurso del tiempo.

De manera más específica un modelo de datos “es un conjunto de conceptos, reglas y convenciones que nos permiten escribir y manipular (actualizar o consultar) los datos de un cierto mundo real que deseamos almacenar en la base de datos”

Y en este momento podemos destacar la diferencia de los modelos con respecto a los lenguajes en este último con características de abstracción ya que se conforman mediante el modelo y una sintaxis. Así la existencia de diferentes lenguajes que pueden introducirse tanto en el modelo como en la sintaxis,

1.3.3. DEFINICIÓN DEL MODELO DE DATOS

Existen tres niveles para representar los modelos¹⁴

Nivel Bajo

Es el que representa los datos de la realidad. Por ejemplo:

Pedro Salazar Ruiz
Calle 24 No.2 Col Congreso 5698748

Características

- 1 Es un registro de la realidad , de datos verdaderos
- 2 Los datos pueden cambiar

Segundo Nivel

Se representa el esquema que describe la estructura de las bases de datos
Como un conjunto de modelos, características que se tienen en común:

- 1 El esquema modela un enorme rango de estados de las bases de datos, ejemplo:
Nombre, Apellido Paterno, Apellido Materno, Teléfono, Dirección, Número de Cuenta.

¹³ De Miguel Adoracion, Plattin Mario Fundamentos y Modelos de Bases de Datos España Ed RA-MA pags 85-86
¹⁴ Haasem W Gary Hansen y Diseño y Administración de Bases de Datos España Ed Prentice-Hall pag 87

Tercer Nivel

Las metodologías describen las construcciones y reglas que pueden ser utilizados en la formulación de un esquema.

Una metodología que estará representada por el modelo:

- 1 Relacional
2. Red
- 3 Jerárquico
- 4 Orientado a Objetos

1.3.4. MAS SOBRE EL MODELO ORIENTADO A OBJETOS

Representados por objetos con su propia identidad y atributos.

Ventajas

- Representan de manera mas precisa la esencia lógica de las aplicaciones del mundo real.
- Por está razón también podría llamarse Modelado Semántico
- Ya que nos proporciona un medio poderoso para asociar el significado de las cosas de la realidad.
- Las bases de datos orientadas a objetos han sido desarrolladas bajo un modelo y lenguajes orientados a objetos

Conceptos Básicos

- Objeto-Identidad
Cada entidad del mundo real se modela como un objeto
- ┘ Objetos Complejos
Variables de instancia asociados a un objeto, lo que permite crear objetos en términos de otros objetos.
- Encapsulamiento
Cada procedimiento define métodos y la manera en que pueden acceder a estos.
Relacionado con los métodos y atributos
- ┘ Clases Objetos
Cada objeto comparte un mismo atributo
- Herencia Una clase puede heredar atributos, el nodo padre será la superclase en la que se derivan todos los conceptos de subclases.
- ┘ Sobrecarga(Overloading) con esta función pueden asociarse diferentes métodos a un mismo nombre de operación.

Y con respecto al desarrollo de las bases de datos orientados a objetos, se ven involucradas:

- Objetos Compuestos

- Versiones
- Abstracciones que van mas ligadas al mundo de los lenguajes.
- Construcción de anidamiento de datos; objetos que se refieren a otros objetos
- Utilización de punteros, entre los identificadores de objetos.

Los modelos orientados a objetos, no tienen una teoría matemática y esto lo atribuyen al álgebra o cálculo relacional.

Pero que ni el álgebra ni el cálculo manejan aspectos como la concurrencia, la recuperación o control de integridad.

En el paradigma¹⁵ de la orientación a objetos, un sistema se considera como un conjunto de objetos que se comunican entre sí mediante mensajes.

1.4. PROPIEDADES GENERALES DE LA ORIENTACION A OBJETOS

- Encapsulamiento
- Modularidad
- Jerarquía

1.5. ELEMENTOS DE ORIENTACIÓN A OBJETOS

- Objetos
- Clases
- Métodos
- Mensajes
- Herencia
- Polimorfismo

1.5.1. OBJETOS

- Es una entidad que contiene los atributos que describen el estado de un objeto del mundo real y las acciones que se asocian con el objeto del mundo real
- Tiene una interfaz público y un interfaz privada
- Son instancias de una clase.
- Consta de funciones, números, arreglos, cadenas, registros y subrutinas
- Las Funciones: Son el único medio de acceder a los datos privados de un objeto si se desea leer un dato de un objeto, se llama a la función de un objeto no se puede acceder a los datos directamente, ninguna otra función puede alterar los datos, lo que simplifica escritura, depuración y mantenimiento

¹⁵ El conjunto de teorías, estructuras y métodos que juntos representan un medio de visualizar al mundo

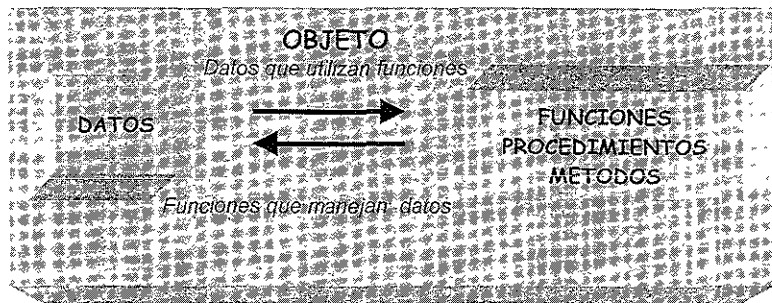


Figura 2.3 Representación de un objeto.

La estructura interna de un objeto consta de dos componentes básicos¹⁶

- Atributos
- Métodos(operaciones o servicios)

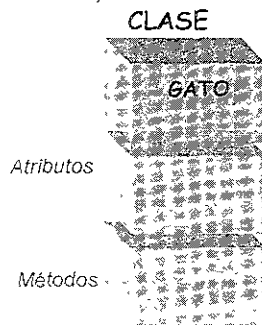


Figura 2.4 Notación gráfica OMT de una clase y de un objeto.

Ejemplo de Objetos:

- Objetos Físicos
 - Aviones
 - Carros
 - Arboles
 - Trenes
 - Casas
- Animales
 - Vertebrados
 - Invertebrados
 - Pescados
- Elementos de interface gráficos
 - Ventanas

¹⁶ Jovanes Aguilar Luis Programación Orientada a Objetos España Estructura Interna de un Objeto pag 87

- Menús
- Cuadros, triángulos
- Teclado
- Tipos de datos definidos por el usuario
 - Puntos de un sistema coordinado
 - Datos Complejos

1.5.2. MENSAJES

Un mensaje es una petición de un objeto a otro objeto al que le solicita ejecutar uno de sus métodos. Por convenio el objeto que envía la petición se le denomina emisor y el objeto que recibe la petición se denomina receptor.

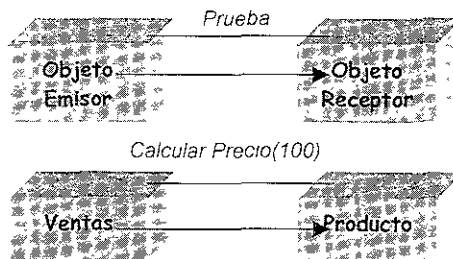


Figura 2.6 .Objeto emisor y receptor de un mensaje.

1.5.3. CLASES

Es la descripción de un conjunto de objetos, consta de métodos y datos que resumen características comunes de un conjunto de objetos.

- Declaración de un tipo de objetos
- Son similares a los tipos de datos y equivalen a modelos o plantillas que describen cómo se construye cierto tipo de objetos
- Los objetos son instancias¹⁷ de una clase.

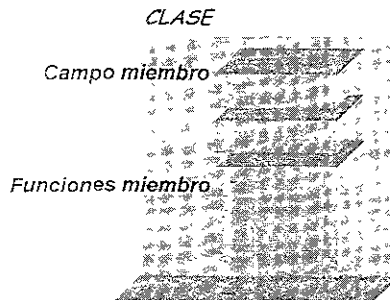


Figura 2.7 Diagrama de una Clase

¹⁷ Se construye un objeto a partir de una clase

1.5.4. HERENCIA

La encapsulación es una característica muy potente, junto con la ocultación de la información, para describir mejor este punto se mencionan lo siguiente:

- Representa un concepto avanzado de datos
- Adquiere mayor relevancia cuando se encapsulan e integran datos
- Se incorporan características de herencia.
- Construcción de objetos a partir de otros.
- Su objetivo es la reutilización, es decir utilizar código anterior ya desarrollado
- Comparten características comunes.

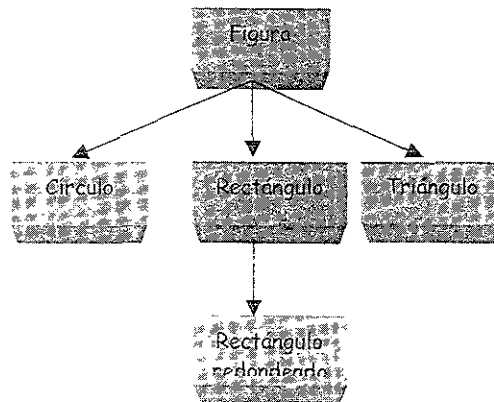


Figura 2. 8 Herencia Simple

1.5.5. POLIMORFISMO(MUCHAS FORMAS)

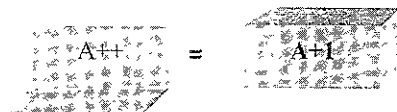
Esta propiedad se encuentra en la mayoría de los lenguajes de programación orientados a objetos

Por ejemplo.

En el lenguaje C++ el símbolo "+"

Tiene dos significados:

1. Puede tomar un proceso aritmético, una simple suma
2. Puede servir para agregar implícitamente un valor



La variable es declarada entera

Cuando se realiza un modelo orientado a objetos, las clase y objetos junto a sus relaciones son los principales elementos de modelado. El modelo de clases y el modelo de objetos muestran el sistema que se trata de describir.

Una vez que el modelo se ha construido y se disponen de clases y relaciones entre ellas, el modelo se lleva a código real; pero llegar a la etapa de implementación y mantenimiento requieren de una buena construcción del modelo.

Es por esta razón que el análisis del sistema requiera metodologías como las descritas a continuación para la implementación de un sistema.

1.6. MÉTODOS DE ANALISIS ORIENTADOS A OBJETOS

Una de las características para poder desarrollar por completo un proyecto y contar con una planeación consistente dentro de un sistema de cómputo, es establecer una metodología enfocada al problema que se desea abordar en este caso la orientación a objetos dentro de las bases de datos, cabe recalcar que con estas herramientas simplifican el nivel de comprensión, establecen reglas de integridad, proporcionan diagramas específicos y detallados, cuentan con una notación propia y en muchos casos son automatizados en herramientas de cómputo

Por lo tanto en las siguientes páginas se describen algunas de éstas metodologías que complementan el desarrollo orientado a objetos y se hace énfasis en alguna de éstas, así como se mencionan muchas más ya que establecen la continuidad de un proyecto de cómputo orientado a objetos.

Por otro lado tómesese en cuenta que éstas metodologías nos ayudan a establecer las reglas para seguir en el desarrollo de un sistema de cómputo enfocado a la orientación a objetos, pero no para la formación del modelo de datos orientado a objetos unidad. cimiento, base fundamental para la creación de una base de datos

1.6.1. MÉTODOS DE ANÁLISIS

Se han desarrollado diferentes metodologías que nos ayudan a comprender el manejo de los objetos, que es el principio mismo de éste método: los objetos, sus atributos con sus respectivos métodos que integrados unos con otros llegan a formar un sistema con la ventaja que pueden tomar diferentes caminos. Entre las metodologías de análisis tenemos¹⁶:

1. OOSA de Shlaer/Mellor
2. Coad/Yourdon
3. Rumbaugh – OMT
4. Martin/Odell – Ptech
5. Objectory y OOSE
6. OORASS

¹⁶ Graham Ian Metodos Orientados a Objetos EU Addison Wesley pags 291-352

7. Desfray – Método de relaciones de clases
8. OSA
9. Systems Engineering OO
10. Texel
11. BON – Nerson
12. Fusion-Coleman
13. SOMA

Enseguida se describirán algunas de éstas metodologías y sus características más importantes:

1.6.1.1. COAD/YOURDON

Características.

- Creado en los años 1990,1991
- Se debe mucho a la tradición de modelados entre entidades y relaciones
- Uno de los primeros en describir ampliamente el método de análisis, con una notación completa, práctica, y orientada a objetos.
- Notación menos torpe que la presentada por Booch y Shlaere/Mellor
- Los atributos resultan completamente explícitos
- Los objetos no solamente se especifican mediante métodos.
- Consta de herramientas para sus notación(Object International OOA Tool).

Para Coad y Yourdon sugieren que el análisis se produce en cinco fases, a las que dan los nombres siguientes¹⁹

- 1 Temas: El área problema se descompone en “temas”, que corresponden con la noción de ‘niveles’ o “capas” en los diagramas de flujo de datos y que equivalen, también a los subsistemas o categorías de clases de Booch

Los temas se representan mediante cajas cuadradas y el paso de mensajes mediante flechas, como si las cajas fueran objetos(fig 2.10).Sin embargo, en este método los temas son descomposiciones arbitrarias que sirven de ayuda para comprender y que no tienen una semántica clara

Algunos temas surgen de forma bastante natural, a medida que los objetos se ensamblan en estructuras de clasificación o de composición



Figura 2.10. Las capas son grupos de, idealmente, entre cinco y nueve objetos.²⁰

Los temas deberán de ser de tamaño tratable, se sugiere que aproximadamente contengan entre cinco y nueve objetos.

¹⁹ Graham Ian. Metodos Orientados a Objetos E U pags 293-301

²⁰ Grupos que representan las capas principales del sistema de Anonno p.144 U Retra

2. Objetos: A continuación se identifican los objetos con detalle.
3. Estructuras. Se identifican dos estructuras completamente distintas: estructuras de clasificación y estructuras de composición, en esta fase se trata la herencia, ya que las estructuras de clasificación son meramente árboles de especialización y generalización.
4. Servicios: En esta fase se realizarán las operaciones. Cada objeto debe contar con métodos para crear y borrar *instancias*, para tomar y poner valores, también se debe disponer de métodos especiales que caractericen el comportamiento del objeto.

En esta fase los métodos o servicios especifican los aspectos funcionales del sistema.

El método Coad/Yourdon resulta especialmente poco adecuado cuando se trata de especificar la dinámica de sistemas.

Otros aspectos importantes de este modelo es su notación la cual cuenta con dos aspectos de las relaciones: multicidad y modalidad

1. Multicidad. Se refiere a que si las relaciones entre objetos pueden ser
 - Uno-a-uno
 - Uno-a-muchos
 - Muchos-a-muchos
2. Modalidad. Se refiere a que si sus objetos pueden ser:
 - Necesarios
 - Posibles(opcionales)

Como podemos ver este modelo está influenciado con la metodología de las bases de datos relacionales, insisten en que los atributos deberían de ser atómicos, siendo esta una desventaja, ya que la orientación a objetos trata de modelar objetos complejos que no tiene por que ser atómicos.

Para entender mejor el método Coad/Yourdon, a continuación se describen de manera resumida el proceso de análisis²¹.

- Identificar objetos usando el criterio de <que buscar>.
- Definir una estructura de generalización - especificación.
- Definir una estructura de todo - parte.
- Identificar temas (representaciones de componentes de subsistemas)
- Definir atributos.
- Definir servicios.

Otro enfoque que al método Coad/Yourdon se le ha dado y que no solamente incluye aspectos de programación o de sistemas, y que pueden ser aplicados a toda una infraestructura, se describen a continuación:

²¹ Preechan Roger S Ingeniería del Software, un enfoque práctico España Ed Mc Graw Hill, pags 388-389

Componente del dominio del problema:

- Agrupar todas las clases específicas al dominio
- Diseñar una jerarquía de clases apropiada para las clases de aplicación
- Trabajar, cuando sea apropiado, para simplificar la herencia
- Refinar y añadir objetos a bajo nivel, en caso de ser necesario.
- Revisar el diseño y proponer adiciones al modelo de análisis.

Componente de interacción:

- Definir los actos humanos
- Desarrollar escenarios para las tareas
- Diseñar una jerarquía de órdenes de usuario.
- Refinar la secuencia de interacción del usuario.
- Diseñar las claves relevantes y la jerarquía de clases.
- Integrar adecuadamente las clases de IGU (Interfaz Gráfica de Usuario)

Componentes para la gestión de datos:

- Diseñar las estructuras de datos y su distribución.
- Diseñar servicios para manejar la estructura de datos.
- Identificar las herramientas que puedan ayudar en la implementación de la gestión de datos.
- Diseñar clases apropiadas y la jerarquía de clases

1.6.1.2. RUMBAUGH – OMT

Características²²:

- Se le considera como uno de los métodos mas completos de análisis OO mas completo.
- Creado en el año de 1991 por James Rumbaugh
- Tiene fuertes raíces de los métodos estructurados tradicionales y ofrece una notación extremadamente rica, pero demasiado complicada y detallada.
- Consta de herramientas automatizadas para su notación
- El método es independiente de los lenguaje.
- Capacidad para representar composiciones recursivas.
- Tiene la intención de ser un método tanto de análisis como para el diseño
- Complejo utilizar sus notaciones
- Limitaciones
- Modelar reglas de negocio
- Control de reglas

Para OMT consta de tres fases o actividades principales:

²² Cíahim Ion Metodos Orientados a Objetos L. U. Ed Addison Wesley pags 301-314

1. Análisis: Presupone que existe una especificación de los requisitos y se desarrolla construyendo tres modelos distintos mediante el uso de tres notaciones diferentes(A, B y C)

A

Modelo Objeto(OM) ,consta de diagramas similares a los de Coad/Yourdon y de un diccionario de datos.

Notaciones del Modelo Entidad/Relación , con operaciones y otras anotaciones que se han añadido.

B

Modelo Dinámico(DM), consta de STD dibujados.

Notación Harel Extendida y diagramas de flujo globales

C

Modelo Funcional(FM), se incluyen a los modelo OM, DM

2. Diseño de Sistemas se realiza organizando los objetos en subsistemas, identificando la concurrencia a partir de DM, asignando a procesadores o tareas, decidiendo si los datos deben o no ser estar almacenados en archivos, en memoria o en un sistema de bases de datos, decidiendo en el uso de periféricos y de recursos globales, escogiendo las estructuras de control, estableciendo las condiciones de contorno(principio/fin/fallo) y estableciendo el equilibrado de prioridades
3. Diseño de Objetos: implica transformar la información del DM y del FM en operaciones de OM y consiste en:

- ▣ Diseñara Algoritmos
- ▣ Optimizar vías de acceso
- ▣ Realizar el control
- ▣ Ajustar estructuras.
- ▣ Indicar los detalles de los atributos
- ▣ Empaquetar las estructuras en módulos.
- ▣ Escribir el informe de diseño, incluyendo un OM, DM y FM detallados
- ▣ En el modelo OMT nos ofrece algunas ideas prácticas para la realización de diseño orientado a objetos en bases de datos relacionales:
- ▣ Construir la identidad de objetos utilizando claves primarias, preferiblemente sustitutos generados por la máquina
- ▣ Las clases son tuplas de tablas e instancias.
- ▣ Las asociaciones son tablas
- ▣ Los enlaces de herencia son ID compartidas con un índice secundario

En el método OMT existen dos caminos diferentes de abstracción

1. El diseño de sistema se centra en la distribución de los componentes necesarios para construir un producto o sistema completo
2. El diseño de objetos enfatiza la distribución detallada de un objeto individual

En las siguientes líneas se describe de manera general el Método de OMT²³.

1. Realizar un diseño del sistema
2. Particionar(partirlo)el modelo de análisis en subsistemas.
3. Identificar la concurrencia dictada por el problema
4. Asignar subsistemas a procesadores y tareas
5. Elegir una estrategia básica para la implementación de la gestión de datos.
6. Identificar recursos globales y los mecanismos de control requeridos para acceder a ellos.
7. Diseñar un mecanismo de control apropiado para el sistema
8. Considerar cómo deben manejarse las condiciones límite.
9. Revisar y considerar intercambios.
10. Conducir un diseño objeto
11. Seleccionar operaciones del modelo de análisis
12. Definir algoritmos para cada operación
13. Seleccionar la estructura de datos apropiadas para los *algoritmos*
14. Definir todas las clases internas
15. Revisar la organización de clases para optimizar el acceso a los datos y mejorar el rendimiento computacional.
16. Diseñar los atributos de la clase
17. Implementar mecanismos de control definidos en el diseño del sistema
18. Ajustar la estructura de clases a una herencia fuerte
19. Diseñar el intercambio de mensajes para implementar relaciones entre objetos.(asociaciones).
20. Empaquetar las clases y asociaciones en módulos

1.6.1.3. NOTACIÓN DE LA METODOLOGÍA ORIENTADA A OBJETOS.

El UML(Lenguaje de Modelado Unificado) es un lenguaje para especificación, visualización y construcción de artefactos(objetos) de los sistemas de software, utilizando conceptos orientados a objetos.

No es un método de análisis y diseño orientado a objetos, si no una notación general y simple que sirve para utilizarse en métodos de desarrollo de software.

Características

- Desarrollado por Grady Booch y James Rumbaugh en el año de 1994.
- Es una notación estándar para modelado de objetos

1.7. CLASIFICACIÓN DE LAS BASES DE DATOS ANTE LA ORIENTACIÓN A OBJETOS

De acuerdo con nuestras metodologías que se han descrito anteriormente, Yourdon/Coad y OMT hacen referencia a un tipo de bases de datos con frontal del

²³ Pesman Roger S Ingeniería del Software, un enfoque práctico España Ed. Mc Graw. pags 411

modelo Entidad/Relación(E/R), pero sin olvidarse de la OO, para definir mejor este concepto se realizan las siguientes clasificaciones de bases de datos frente a la orientación a objetos.

1.7.1. BASES DE DATOS RELACIONALES PURO

En donde se encuentran actualmente algunas bases de datos, y bajo que manejadores de bases de datos están desarrollando.

Empresa	Oracle	DB2	SQL/Server
PEMEX	X		X
PEPSI	X		
IFE	X		
PROCESAR	X	X	X
INE	X		

Figura 2.11. Manejadores de Bases de Datos.

Siendo los manejadores antes mencionados mas importantes en el mercado y con un desarrollo de muchos años, bajo el modelo Entidad Relación.

Existen otros en el mercado que son de menor importancia pero que fueron elaborados bajo el modelo Entidad-Relación (E/R): INGRES, RDB, SYBASE, INFORMIX, INTERBASE.

1.7.2. BASES DE DATOS RELACIONALES CON FRONTAL OO

Se describen por que utilizan conceptos orientados a objetos bajo un modelo E/R, y en donde podemos encontrar con el producto Open ODB²⁴ de HP²⁵

Pero el problema de este enfoque²⁶ es que, al no poder modificar normalmente la base de dato relacional para atender los requerimientos de la metodología OO, se pueden dar problemas.

- 1 Rendimiento
2. Operacionales

Aunque exista la ventaja de la portabilidad de pasar los modelos E/R a OO

1.7.3. BASES DE DATOS ORIENTADAS A OBJETOS PUROS

Estos manejadores soportan únicamente metodologías orientadas a objetos, pero que no han sido mencionados como herramientas administradoras de datos y almacenamiento de información en empresa del Distrito Federal. Entre sus manejadores mas conocidos están O2. Objectivity, Object Store, Ontos, Itasca, Versant, Jasmine, entre otros.

²⁴ Por sus siglas Abmr Base de Dato Objeto (OpenODD)

²⁵ HP -Hawlett Packard

²⁶ Adoracion De Miguel, Plattini Mario Fundamentos y modelos de bases de datos España Ed RA-MA, pags 408-4.1

1.7.4. BASES DE DATOS CON SQL

Estas bases de datos siguen un enfoque unificador, porque combinan OO y el modelo E/R, diferenciándose de las bases de datos con frontal OO, en que se realiza una adaptación a nivel de almacenamiento y a nivel de datos, y un manejador que ofrece estas características es el UNiSQL/X de la empresa UniSQL.

1.8. EXTENSIÓN DE LOS MODELOS ORIENTADOS A OBJETOS.

Tomando como referencia los cuatro tipos en donde se encuentran las bases de datos frente a la OO, la posibilidad de dejar el modelo E/R van desde:

Ampliando el modelo de datos, en donde se involucren los conceptos de OO.

Modificar tanto su modelo como su nivel interno de acuerdo a un enfoque orientado al objeto, más purista, esta postura, será la que se tomará en el trabajo de investigación en cierto momento será optimizada la base de datos, cabe mencionar que se vuelve complicada por el volumen actual y el número de transacciones o movimientos que se realizan sobre la base de datos.

CONCLUSIONES

- De acuerdo al punto 1 se deduce que los objetos contribuyen a mejorar los sistemas de computo, pero para profundizar mas, es necesario, conocer el modelo conceptual que representa la orientación a objetos
- El punto 1.3 al 1.3.3 no se puede partir de tan solo una idea general para diseñar una base de datos sin tomar en cuenta el MODELO, el cual nos ayudará a establecer las bases para la construcción de una base de datos, sin importar si la base de datos es entidad/relación red, jerárquica u orientada a objetos, lo que si es importante, es saber convivir con la funcionalidad de la orientación a objetos y la forma de estructurar el modelo conceptual, partiendo de que se haya analizado de tal forma que la aplicación se apegue lo mas posible a la realidad
- En el punto 1.5 aunque se describe los elementos de la orientación a objetos, aun en nuestros días, cada libro dedicado a esto pretende describir de manera muy particular la forma de visualizar la orientación a objetos y no se contempla la idea de establecer una estandarización de los conceptos, que llevaría a la mayor utilización de la orientación a objetos
- Todo lo contrario al MODELO que establece de manera clara las bases conceptuales, el único inconveniente es la interrelación con los objetos
- Las metodologías de análisis y diseño que se describen en el punto 1.6 enfocados a la orientación a objetos son conceptos abstractos que por su complejidad, no han sido establecidos de forma masiva en las consultorias
- El auge de las bases de datos orientadas a objetos, si es que hay, es hablar de nuevas formas de pensar de nuevas formas de desarrollar, de imponer o de adquirir la responsabilidad de mantener una orientación a objetos, lo cual ha llevado a clasificar las bases de datos con demas teorías como lo es la orientación a objetos y la relacional, y se han desarrollado bases de datos con funcionalidad tanto de una como de otra, por lo cual estamos en una transición entre usar un modelo relacional orientado a objetos hacia un modelo orientado a objetos de forma pura.
- Habrá un momento en la historia de la computadora y del software y por supuesto del hardware que por el simple hecho evolutivo, pasaremos a desarrollar de forma orientada objetos o simplemente objetos

2. PLANTEAMIENTO DE DIFERENTES MODELOS EN COMPARACIÓN CON EL MODELO DE DATOS ORIENTADOS A OBJETOS, EN CONJUNTO CON LAS BASE DE DATOS.

2.1. MODELO ENTIDAD-RELACIÓN

El modelo entidad-relación ha sido establecido por muchas empresas en sus bases de datos, es hasta el momento el modelo con mayor número de aceptación por parte de los desarrolladores, y ha sido reutilizado por IBM, HP, ORACLE, Informix, entre otros

Pero a que se debe que su funcionalidad haya traspasado por todo el análisis y diseño de software y hardware; gracias a su operatividad, gracias a su metodología a sus conceptos y teorías, en conjunto se puede resumir que si, pero para la comunidad científica y para la aceptación por parte de la inversión computacional, el modelo de datos entidad-relación ha sido establecido por que cuenta con bases científicas, como lo es el álgebra relacional, bajo este concepto se han propuesto bases completamente demostrables y que solo se comunican con el ámbito computacional desarrollando herramientas de diseño, manejadores, ahora la pregunta, ha llegado a su fin éste modelo, o solo por el hecho de que el número de avances científicos computacionales avanzan a pasos agigantados la utilización del modelo (E-R) no deberá de utilizarse mas, y ha sido reemplazado por el modelo de datos orientados a objetos será factible, será preciso contar ya con otro modelo, sin importar si es o no el momento de cambiarlo, primero habiemos de los conceptos básicos del modelo entidad-relación.

Características:

Existen tres nociones básicas que emplea el modelo de datos entidad-relación²⁷:

²⁷ Subscritor Abraham Korchi Hea y J. Sridarshan Fundamentos de bases de Datos España Ed. Mc Graw Hill pags 15-19

- ┌ Conjunto de Entidades
- ┌ Conjunto de Relaciones
- ┌ Atributos

Conjunto de entidades

Entidad: es una COSA u OBJETO en el mundo real que es distinguible de todos los demás objetos

Por ejemplo:

Podemos decir que una persona es una entidad

Una entidad tiene un conjunto de propiedades, y los valores para algún conjunto de propiedades puede identificar de una entidad de forma unívoca

De manera mas descriptiva una entidad es "una persona, lugar, cosa, concepto o suceso real o abstracto, de interés para una empresa". Es aquel objeto en el cual queremos almacenar información en la base de datos, Por ejemplo el RFC, NSS²⁸ identifica de manera unívoca a una persona. Una entidad puede ser concreta como un libro o flor o puede ser abstracta, como un préstamo, vacaciones o un concepto(véase fig.2 1)

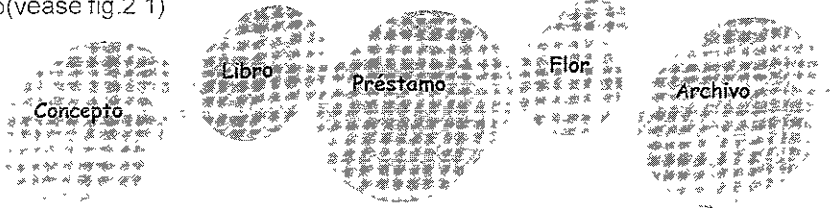


Figura 2.1 Representación de Entidades.

Por lo tanto un CONJUNTO DE ENTIDADES es la totalidad de las entidades del mismo tipo que comparten las mismas propiedades o atributos.

El conjunto de todos los empleados afiliados a un AFORE²⁹, se puede definir como el conjunto de entidades AFILIADOS.(véase fig. 2 2)

Entidad: AFILIADOS

Nombre	RFC	NSS	Dirección	Fecha Nacimiento	Edad

Figura 2.2 Entidad AFILIADOS

De manera similar podemos definir el conjunto de todas las AFORE, se pueden definir como el conjunto de AFORE.(véase fig. 2 3)

²⁸ NSS Num. de Seguro Social asignado por el IMSS
²⁹ AFORE Administradores de Fondo para el Retiro

Entidad: AFORE

Nombre	Identificador	Dirección	Fecha Nacimiento	Clave Afore

Figura 2.3 Entidad AFORE

Las entidades individuales que constituyen un conjunto se llaman la extensión del conjunto de entidades.

Los conjuntos de entidades no son necesariamente disjuntos. Por ejemplo, es posible definir el conjunto de entidades de todos los empleados de una empresa (EMPLEADO) y el conjunto de entidades de todos los clientes del banco (CLIENTE). Una entidad persona puede ser una entidad EMPLEADO, una entidad CLIENTE, ambas cosas o ninguna.

Una entidad se representa mediante un conjunto de atributos. Los atributos describen propiedades que posee cada miembro de un conjunto de entidades. La designación de un atributo para un conjunto de entidades expresa que la base de datos almacena la información similar concerniente a cada entidad del conjunto de entidades, sin embargo cada entidad puede tener su propio valor para cada atributo. Ejemplo (véase fig 2.4)



Entidad: AFILIADOS

Nombre	RFC	NSS	Dirección	Fecha Nacimiento

Entidad AFORE

Nombre	Identificador	Dirección	Fecha Nacimiento

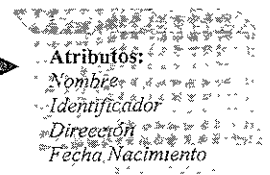


Figura 2.4. Representación de Atributos en Entidades.

Para cada atributo hay un conjunto de valores permitidos, llamados el dominio o el conjunto de valores permitidos, de ese atributo. En el ejemplo anterior podemos decir que el atributo NOMBRE de la entidad AFILIADOS podría ser el conjunto de todas las cadenas de texto de una cierta longitud, o bien el atributo ID de la entidad AFORE podría ser el conjunto de todos los enteros positivos.

Por lo que una base de datos representa una colección de conjuntos de entidades, cada una de las cuales contiene un número de entidades del mismo tipo (véase fig 2.5)

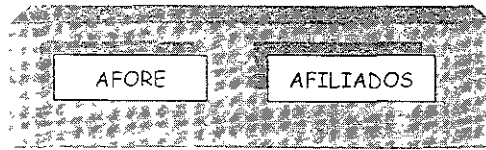


Figura 2 5 Base de Datos AFORE con algunas Entidades.

De manera formal, un atributo de un conjunto de entidades es una función que asigna al conjunto de entidades un dominio. Como un conjunto de entidades puede tener diferentes atributos, cada entidad se puede describir como un conjunto de pares (atributos, valor), un para cada atributo del conjunto de entidades. Por ejemplo una entidad concreta como AFILIADOS se puede describir mediante el conjunto NOMBRE, RFC, NSS DIRECCION, FECHA DE NACIMIENTO (véase fig. 2.6)

Entidad: **AFILIADOS**

Nombre	RFC	NSS	Dirección	Fecha Nacimiento
Ricardo	PERR790612	22223145697	Sur 234 No 1247 Col Ejercito	12-Jun-1979
Tania	HERT760506	32632632611	Azul Gomez No 124 Col Granjita	08-May-1976
Pedro	GOPF680103	76986955896	Florida No 589 Col Cuadra2	13-Jan-1968

Figura 2.6 Representación de Entidad AFILIADOS con sus atributos.

Lo que significa que una persona de la entidad AFILIADOS describe una persona con NOMBRE Ricardo con un RFC PERR790612, con un NSS 22223145697 con una DIRECCION Sur 234 No 1247Col Ejercito y con una FECHA DE NACIMIENTO 12-Jun-1979 Los valores de los atributos que describen una entidad constituirán una porcion significativa de los datos almacenados en la base de datos

Un atributo puede clasificarse entre los siguientes tipos:

- Simples y Compuestos
- Univalorados
- Multivalorados
- Nulos
- Derivado

Conjunto de Relaciones

Una relación es una asociación entre diferentes entidades. Por ejemplo, se puede definir una relación que asocie al afiliado Ricardo con la AFORE Inbursa. Esta Relación específica que Ricardo es un afiliado con la AFORE Inbursa. (véase fig. 2.7)

Nombre	RFC	NSS	Dirección	Fecha Nacimiento
Ricardo	PERR790612	22223145697	Sur 234 No 1247 Col Ejercito	12-Jun-1979



Entidad: AFORE

Nombre	Identificador	Fecha Nacimiento
INBURSA	036	10-Apr-200

Figura 2.7 Relación de Entidades.

Relaciones

Expresa el número de entidades a las que otra entidad puede estar asociada vía un conjunto de relaciones, esta relación describe conjunto de relaciones binarias, aunque ocasionalmente contribuye a la descripción de conjuntos de relaciones que implican mas de dos conjuntos de entidades

Para un conjunto de entidades binarias R entre conjuntos de entidades A y B la correspondencia de cardinalidad debe de ser una de las siguientes (véase fig. 2.8)

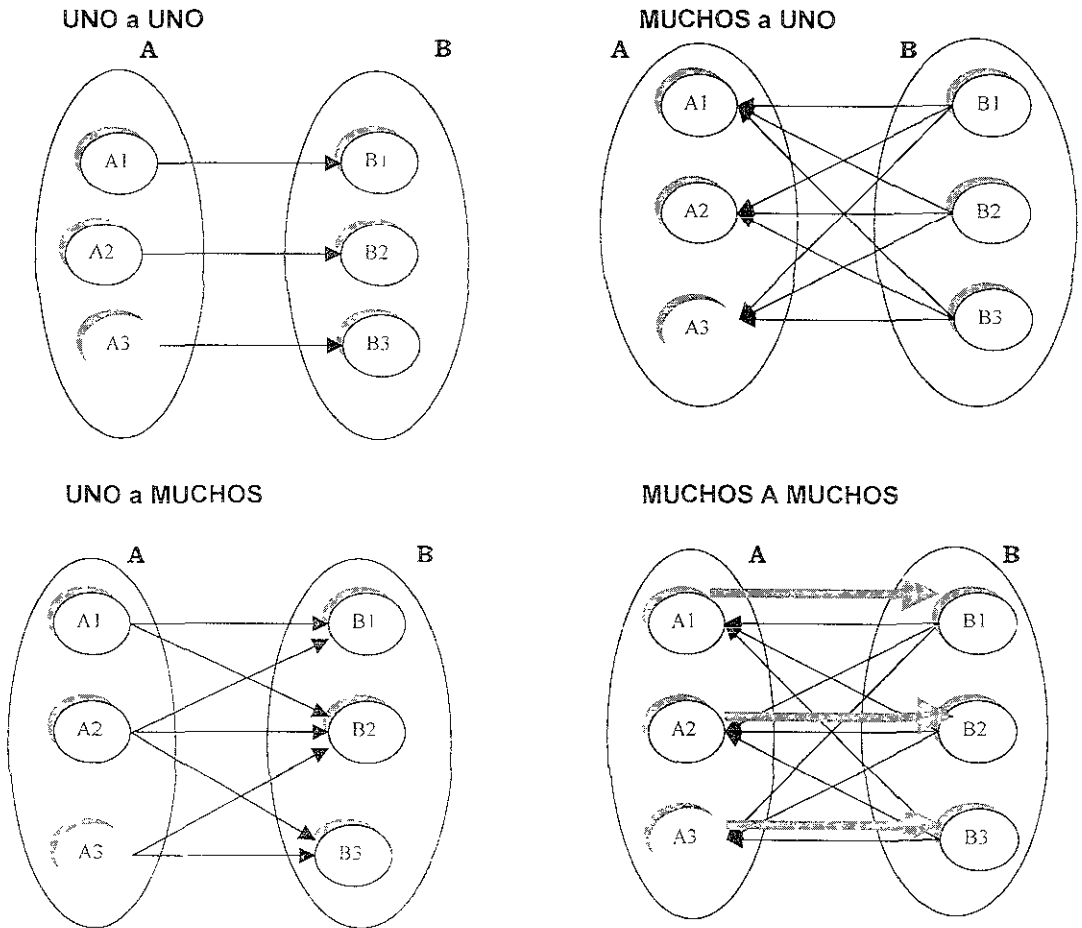


Figura 2.8 Relaciones

Mencionado los conceptos elementales del modelo entidad-relación (E-R) . de las características de la orientación a objetos, de la descripción del modelo enfocado al desarrollo de bases de datos, de las metodologías de análisis se va conformado todo una serie de elementos que integran por completo el objetivo de la investigación, por otro lado es conveniente establecer otro tipo de estudios que de manera retrospectiva ayuden a los diferentes lectores a establecer líneas de investigación de menor escala pero de mucha importancia

2.2. ESTUDIOS PARA IMPLEMENTAR UNA BASE DE DATOS ORIENTADA A OBJETOS

Para este caso se mencionarán diferentes estudios:

Del entorno

En este caso se trata de investigar cual es el alcance que han tenido las bases de datos orientadas a objetos, y si en realidad hay aplicaciones ya en el mercado. Que metodologías han utilizado y cual es el modelo de datos que han utilizado.

De sistema y procedimientos

Se verifica cual es el proceso para la creación de una base de datos de acuerdo a que si la base de datos esta desarrollada bajo un Modelo Entidad-Relación o un Modelo Orientado a Objetos.

De conservación y mantenimiento

Siguiendo con las reglas del negocio, una de sus prioridades es, conservar su estructura del sistema con un adecuado funcionamiento. El implementar una base de datos orientada a objetos nos daría como pauta un avance que conserve y mantenga la estructura de una base de datos.

De la información

La prioridad de las empresas que cuentan con una gran cantidad de información es cuidarla y mantenerla bajo sistemas de información, y una de estas herramientas que nos proporciona son las bases de datos y si a esto le agregamos una metodología y un modelo orientado a objetos podríamos aumentar la complejidad y estandarizar la creación de una base de datos orientada a objetos, que en el futuro o a largo plazo no necesite de cambios significativos.

2.3. COMPARACIONES CON OTROS MODELOS

Como no existe un modelo orientado a objetos para tomarlo de referencia, a lo largo de varios años los investigadores han propuesto diferentes modelos orientados a objetos por lo tanto si podemos describir y definir diferentes características que puedan llegar a consolidar un Modelo Orientado a Objetos, aunque todavía falta un largo camino por recorrer en comparación con el Modelo Entidad-Relación que se estableció por sus bases científicas, por su álgebra y el cálculo relacional, hasta la fecha existen diferentes modelos que han evolucionado y nos han puesto en el camino de la Orientación a Objetos. Cabe aclarar que la comparación que en un principio se tenía era sobre las metodologías de análisis o diseño orientadas a objetos que son OMT, JOURDON, UML, mencionadas en el capítulo anterior y que son reglas establecidas por los diferentes autores con base a su experiencia para el

desarrollo de sistemas que faciliten la integración de las reglas del negocio, pero también existen modelos con base a diferentes características que los sistemas han requerido, a continuación se describirán los modelos que han dado una pauta para el desarrollo del MODELO DE DATOS ORIENTADO A OBJETOS.

2.3.1. MODELOS SEMÁNTICOS DE DATOS

En este caso se hacen mención a tres modelos semánticos:

- Modelo Entidad-Relación (Chen, 1976)
- Modelo Funcional DAPLEX(Suipman, 1981)
- Modelo Semántico de Datos

Su característica es el obtener o capturar tantos conjuntos de relaciones semánticas entre entidades del mundo real como sea posible, buscan de manera eficiente las relaciones de Generalización/Especialización, Agregación e Instancia-de.

A diferencia del modelo de datos orientado a objetos con el modelo semántico de datos, es menos expresivo, pero este último carece del concepto método.

Otra característica que se debe de agregar al modelo de datos orientado a objetos es incluir funcionalidades tales como las versiones y los objetos compuestos.

Los modelos semánticos ofrecen mecanismos para una abstracción estructural y los modelos orientados a objetos ofrecen una abstracción del comportamiento, por lo tanto es más similar a los lenguajes de programación, aunque los modelos orientados a objetos avanzados ofrecen mecanismos para dar soporte adecuadamente a ambos tipos de abstracción

La figura 2.9 nos mostrará las diferencias mencionadas entre el Modelo de Datos Orientado a Objetos con el Modelo Semántico de Datos.

Características	Modelo de Datos Orientado a Objetos	Modelo Semántico de Datos	Modelo de Datos Orientados a Objetos Avanzados
Capturara conjunto de relaciones semantica del mundo real		X	
Relaciones de Generalizacion/Especializacion		X	
Relacion de Agregacion		X	
Relacion de Instancia-de		X	
Menos expresivo	X		
Carece de metodo		X	
Manejo de Versiones	X		
Objetos Compuestos	X		
Abstraccion estructural		X	X
Abstraccion del comportamiento	X		X

Figura 2.9 Diferencias entre MODELOS.

2.3.2. MODELO DE DATOS EN RED Y JERÁRQUICO

Existen dos tipos de semejanzas entre los modelos de red y los modelos orientados a objetos ,la primera es que permiten alguna forma de anidamiento de datos, lo que quiere decir que se aceptan objetos que se refieren a otros objetos con valores de sus atributos. la diferencia es que la jerarquía de agregación en el esquema de objetos puede contener ciclos, por el contrario la modelación de objetos cíclicos en el modelo de redes requiere que se introduzcan estructuras artificiales en el sistema.

La segunda semejanza que existe, se puede observar en los identificadores de objetos y el uso de punteros en el modelo de redes. Definiendo a un identificador de objeto , podemos decir que es un puntero lógico, y quizá la desventaja que pueden tener los sistemas es que nunca se reutilicen estos identificadores, por lo que se elimina el objeto, mientras que un puntero a un registro ,es un puntero físico y no puede ser, por tanto, utilizado para verificar la integridad referencial.

En la siguiente figura 2 10 se apreciarán mas las características que existen entre los modelos de datos jerárquico y de red y los modelos de datos orientados a objetos

Características	Modelo de Datos de Red y Jerárquico	Modelo de Datos Orientados a Objetos
Anidamiento de datos	X	
Contener Ciclos		X
Estructuras Artificiales	X	
Identificadores de Objetos	X	X
Punteros	X	X

Figura 2.10 Diferencias entre MODELOS.

2.3.3. MODELO RELACIONAL DE DATOS

El modelo relacional difiere del modelo orientado a objetos en que los objetos complejos no se pueden modelar directamente, puesto que los valores de los atributos solo pueden ser primitivos, y que no prevé la noción de herencia. No hay mecanismos para asociar las operaciones definidas por los usuarios con la definición de los objetos en el esquema de la base de datos, y la semántica sobre el comportamiento de los objetos queda dispersa en los programas de aplicación.

El modelo relacional no contempla el concepto de identidad de un objeto como un concepto independiente del estado del objeto.

A continuación en la figura 2 11 se mostrarán las diferencias entre estos dos modelos.

Características	Modelo Relacional	Modelo de Datos Orientados a Objetos
Objetos Complejos		X
Valores Primitivos		X
Noción de Herencia		X
Elaboración de mecanismos propios		X
Semantica sobre el comportamiento		X
Identidad de un objeto		X

Figura 2.11 Diferencias entre MODELOS

2.4. EL MODELO DE DATOS Y LAS BASES DE DATOS

Muchos de los conceptos que hasta ahora se conocen de la orientación de objetos, provienen de lenguajes de programación, SIMULA(1966), que más tarde se ampliará el concepto de objetos bajo el lenguaje de SMALLTALK(1983), desde entonces se han desarrollado otros lenguajes como son C++(1986), CLOS(1989) y Eiffel(1988) El principio básico para considerar un lenguaje programación orientada a objetos es que se componga de objetos independientes, agrupados en clases, los cuales se comunican con otros objetos por medio de mensajes.

Estos conceptos también fueron desarrollados en otras áreas, una de estas fueron las bases de datos

Las cuales requieren de un MODELO DE DATOS propio y en ausencia de un estándar, ciertos conceptos concernientes al modelo de datos se han agrupado en un modelo central (núcleo) o modelo básico. Esta solución es lo suficientemente poderosa para integrar un sistema y satisfacer muchos de los requerimientos de los sistemas avanzados

Una de las restricciones del modelo central es la integridad y las relaciones semánticas que son tan importantes para muchos tipos de aplicaciones. Ejemplos:

Integridad

- Unicidad de los valores de un atributo
- La posibilidad del valor nulo para un atributo
- Rango de valores de un atributo que puede asumir

Relaciones Semánticas

- Son consideradas esenciales
- Incluyen la noción de <<parte de >> entre pares y asociaciones de objetos

2.5. CONCEPTOS BÁSICOS DEL MODELO DE DATOS ORIENTADO A OBJETOS

- Objeto e identidad
- Objetos Complejos
- Encapsulamiento
- Clases

- Herencia
- Sobrecarga

Mucha relación tendrá que ver los siguientes conceptos con los mencionados en el capítulo uno, mas no es lo mismo ya que unos están enfocados hacia la idea purista de la orientación a objetos y otros están enfocados hacia los modelos de datos orientados a objetos. que apesar de contar con similitud de conceptos, el empezar a especificar un modelo es primero establecer las propias bases, únicas, que solo pertenezcan al modelo del que se esta hablando .

En la siguiente figura 2 12 se muestran los conceptos básicos del MODELO DE DATOS ORIENTADO A OBJETOS

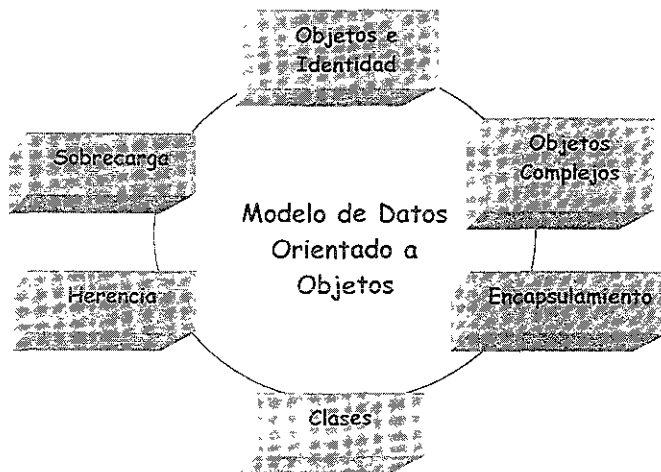


Figura 2.12 Elementos del Modelo Orientado a Objetos.

Cabe señalar que ya anteriormente se había mencionado el Modelo de Objetos, y sus diferentes características, que en lo siguientes descripciones se dará una idea mas específica a lo que se desea llegar en el la investigación, que es la obtención de un MODELO DE DATOS ORIENTADO A OBJETOS EN UNA BASE DE DATOS.

2.5.1. OBJETOS E IDENTIDAD³⁰

En un sistema orientado a objetos, cada entidad del mundo real se representa por un objeto al cual está asociado un *estado* y un *comportamiento*. El estado se representa por los valores de los atributos del objeto El comportamiento se define por los métodos que actúan sobre el estado del objeto cuando se involucran las operaciones correspondientes.

³⁰Benito Elisei y Enzo Marino. Sistemas y Bases de Datos Orientados a Objetos. E.U. Ed Addison Wesley/Diaz de Santos pag 14

Ahora, cada objeto siempre se identifica por un simple IDO(Identificador de Objeto).Por lo que cada objeto existe independientemente de los valores de los atributos del objeto, lo que le da mayor validez , es el uso de objetos que pueden compartir otros objetos y que a su vez se construyen redes de objetos

2.5.2. OBJETOS Y VALORES

En otros modelos en los cuales se permiten objetos y valores(a menudo son llamados literales),y donde no todas las entidades se representan como objetos. Mencionemos las entidades primitivas, tales como enteros o caracteres, se representan por valores, mientras las entidades no primitivas se representan por objetos.

2.5.3. OBJETOS COMPLEJOS

Son un conjunto de atributos (o variables de instancia, o <<slots>>) está asociado a cada objeto, el valor de un atributo puede ser un objeto o un conjunto de objetos. Estas características permite definir objetos arbitrariamente complejos en términos de otros objetos

2.5.4. ENCAPSULAMIENTO

Cada objeto contiene y define los procedimientos(métodos) y la interfaz mediante la cual se puede acceder a él y otros objetos pueden manipularlo. La interfaz de un objeto consiste en un conjunto de operaciones que pueden ser invocadas sobre el objeto. El estado de un objeto(atributos) es manipulado mediante los métodos invocados por las operaciones correspondientes

2.5.5. CLASES

Todos los objetos que comparten el mismo conjunto de atributos y métodos se agrupan en clases. Cada objeto pertenece es una instancia(ejemplar) a alguna clase.

2.5.7. SOBRECARGA

Sobrecarga(overloading), suplantación o anulación (overriding) y ligadura tardía(late binding) con estas funciones se pueden asociar diferentes métodos a un mismo nombre de operación, dejando que sea el sistema quien determine qué método es el que debe ser utilizado para ejecutar una determinada operación

2.6. ¿QUÉ ES UN BASE DE DATOS ORIENTADA A OBJETOS?

De manera orientada a objetos una base de datos se describe de la siguiente manera:

Se han mencionado, que las propiedades de la Orientación a Objetos(OO) son

1. Encapsulamiento la capacidad de enfrentarse a los objetos complejos y de encapsular métodos y estructuras de datos dentro de ellos.
2. Herencia: la capacidad de los objetos para participar en redes y compartir tanto los atributos como los métodos.
3. Identidad de Objetos o de sí mismos

Existen características de los lenguajes de programación OO que no prestan apoyo a las aplicaciones básicas en términos de datos

- ❑ La persistencia de objetos
- ❑ La administración de grandes cantidades de datos
- ❑ La integridad de datos y de transacciones
- ❑ El acceso multiusuario compartido y concurrente
- ❑ Los lenguajes de acceso de consulta
- ❑ La recuperación
- ❑ La seguridad

En primer lugar los objetos pueden ser creados y destruidos en memoria mediante una aplicación (programa) o un compilador .Si alguno de ellos persiste. Existen dos maneras de verificar su almacenamiento

1. Almacenados en una base de datos convencional
- 2 Almacenados como objetos

El primero, significa que los objetos no están preparados para ser utilizados, que se quiere decir con esto, que si un dato es requerido dentro de un programa ,se debe especificar en el programa el nombre del archivo texto con las funciones propias de cada lenguaje, y así físicamente se contará con el campo o con la información solicitada en el programa, a consecuencia de que si no existe tal archivo la función de dicho programa se verá afectada.

Y el segundo ,cuando utilizamos objetos, el objeto se vaciará directamente, por lo tanto un objeto puede estar listo en cualquier momento.

La persistencia de los objetos por lo tanto tendrá que ver con la Identidad de Objetos.

De manera mas detallada un sistema base de datos orientada a objetos³¹ (o modelo de datos) es un sistema de bases de datos(o modelos) en el sentido de que posee todas las características como. Lenguajes de acceso, capacidad de administrar

³¹ G. abram. Ian Métodos Orientados a Objetos E U Ed Addison-Wesley/Diaz s de Santo pag 20*

grandes cantidades de datos, la persistencia, la integridad de datos y de transacciones, la concurrencia, la seguridad y la recuperación, y que adicionalmente apoya la abstracción, la herencia y la identidad de objetos

Identidad de Objetos

La capacidad de un sistema para diferenciar objetos, a lo largo de su duración , no de acuerdo con sus componentes o con sus atributos, si no mediante un identificador único

Técnicas para identificar objetos

Existen tres técnicas convencionales para identificar objetos que son:

1. Por su ubicación física (memoria, punteros)
2. Claves Primarias
3. Identificadores definidos por el usuario

Construcción de identidades de un Objeto

Existen varias formas de construir identidades la mas común es, el uso de un identificador único generado por el sistema.

Características

La identidad de un objeto.

No dependerá , de su:

- Posición
- ┘ Propiedades Cambiantes
- Propiedades No Cambiantes
- Objeto único
- ┘ Relaciones entre objetos
- Uso de expresiones orientados a valores(SQL) (véase fig.2.13)

Ejemplo.

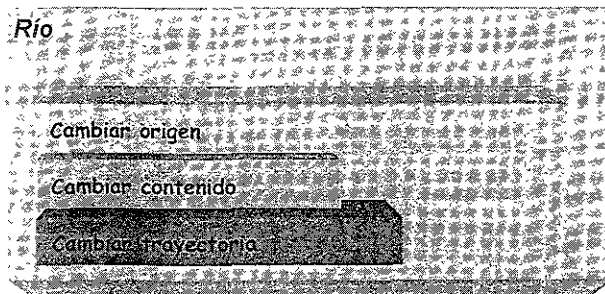


Figura 2.13. Identidad de un Objeto.

Por lo tanto un base de datos orientada a objetos (o modelo de datos).

- ❑ Unifica el acceso basado en valores y basado en identidades.
- ❑ Es un sistema de bases de datos (o modelos)
- ❑ Posee todas las características generales de una base de datos.
- ❑ Lenguajes de acceso
- ❑ Capacidad de administrar grandes cantidades de datos.
- ❑ Persistencia
- ❑ integridad de datos
- ❑ Transacciones
- ❑ Concurrencia
- ❑ Seguridad
- ❑ Recuperación
- ❑ Apoya la abstracción
- ❑ Herencia
- ❑ Identidad de Objetos
- ❑ Su funcionalidad depende en gran parte de los modelos semánticos de datos
- ❑ Modelar propiedades de las estructuras de datos
- ❑ Dificultad para crear estructuras de herencia
- ❑ Clasificación de objetos de acuerdo con su clase

Clasificación

Los objetos pueden ser agrupados en clases de acuerdo con sus propiedades comunes de estructura y de comportamiento.

Conceptualmente un objeto (véase fig 14)

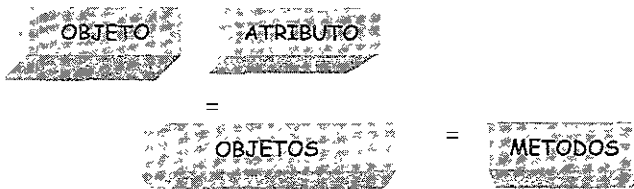


Figura 2.14. Clasificación de Objetos

Los objetos pueden agruparse como

1. Componentes de algún objeto compuesto de dos maneras
2. Como estructura de partes
3. Conjunto de Atributos

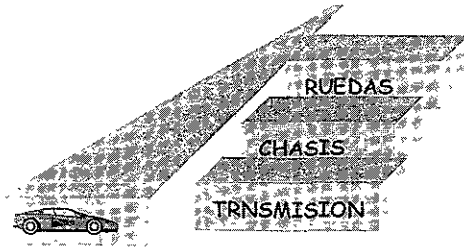


Figura 2.15 Estructura de un OBJETO

2.7. CARACTERÍSTICAS GENERALES DE LAS BASES DE DATOS ORIENTADAS A OBJETOS

- Persistencia
- Modelo de Objetos
- Lenguajes de Datos
- Arquitectura a tres niveles
- Optimizador
- Seguridad
- Interfaces para Aplicaciones
- Diccionario de Datos
- Herramientas para el desarrollo de aplicaciones
- Herramientas para la transferencia de datos
- Utilidades para el mantenimiento y mejora del rendimiento de la base de datos
- Soporte de datos Multimedia
- Herramientas y facilidades de usuarios
- Distribución
- Control de Versiones y Configuraciones.

El modelo en el cual se ha desarrollado la parte de la base de datos en el ejemplo que en el capítulo tres se especificará es un MODELO ENTIDAD-RELACION, un modelo que se han descrito las características básicas que han desarrollado al crear nuevas bases de datos, pero cual es el objetivo, implementar una BASE DE DATOS ORIENTADA A OBJETOS por eso se han mencionado diferentes tecnologías que involucran la ORIENTACION A OBJETOS que revelan el comportamiento de la ORIENTA A OBJETOS, enseguida se hará mención de un modelo que se ha desarrollado desde los inicios de los años noventa y que hasta la fecha sigue desarrollándose para implementar un MODELO, pero que sea aplicable a los OBJETOS y esto implica una descripción mas detallada del modelo

2.8. MODELO ODMG-3

Se mencionará la riqueza del modelo de objetos (MO) en bases de datos puede proporcionar en el lenguaje de consulta. En primer lugar se retoman los elementos del MO y en segunda se muestran ejemplos de uso de los lenguajes del estándar (lenguaje para definición de objetos, lenguajes de consulta, lenguajes para escribir aplicaciones).

Arquitectura

El modelo ODMG-3 busca de manera transparente la INTEGRACION de la base de datos, entre los lenguajes de programación y los sistemas de base de datos que en este caso será de un módulo de TRASPASOS de la base de datos de PROCESAR que antes era parte del Sistema de Ahorro para el Retiro (SAR) que controlaba el IMSS, con la nueva ley y la creación de una empresa que se dedique a administrar, controlar, mantener y procesar la información que concierne al retiro de todos los trabajadores, se crea la empresa PROCESAR, la cual ha creado diferentes módulos para administrar la base de datos.

- └ TRASPASOS
- └ RECAUDACION
- └ AFILIACION
- └ RETIROS
- └ LIQUIDACION

Para este modelo se tomará el módulo de TRASPASOS en su fase de solicitudes, para la inscripción a una AFORE en su parte primera, porque digo primera ya que el módulo de TRASPASOS cuenta con tres satélites, con tres fases que son (véase fig 2.16)



Figura 2.16 Traspasos y sus satélites.

El objetivo principal de este modelo es lograr la compatibilidad global de aplicaciones

- └ Lograr portabilidad de código fuente
- └ No solamente en las pares escritas en el lenguaje de consulta y evitar un lenguaje de manipulación de datos.
- └ Solucionar problemas de desajuste entre la representación de los datos en los lenguajes de programación y en las bases de datos.

Y las ventajas que podemos encontrar serán:

- └ Persistencia de lenguajes orientados a objetos (C++, JAVA)

- Manejo de Transacciones
- Capacidad para manejo de consultas
- Compartir objetos entre aplicaciones, que se ejecuten concurrentemente.

Ahora bien para trabajar con una base de datos orientada a objetos

1. El programador proporciona un programa fuente para su aplicación, además de escribir declaraciones de datos e interfaces para el esquema de la misma
2. El esquema puede escribirse en un lenguaje para definir objetos(ODL)
3. El programa fuente en un lenguaje de programación que tenga extensiones, que se han mencionado en el capítulo uno, para procesar transacciones y consultas.
4. Las declaraciones y el programa fuente se compilan y ligan con el sistema de bases de datos orientadas a objetos, para producir una aplicación ejecutable
5. La aplicación accede a bases de datos nuevas o existentes cuyos tipos forman parte de las declaraciones

Partiendo de que una BASE DE DATOS OBJETO es un modelo de objetos más una base de datos(véase fig 2.17)

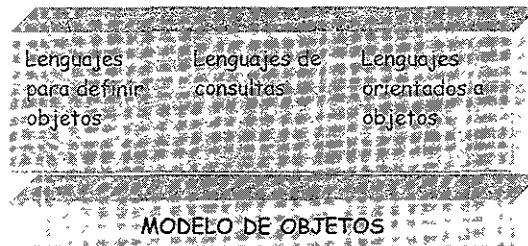


Figura 2.17 Modelo de Bases de Datos. ODMG-3.

- A. Un modelo de objetos, el cual se basa en el modelo de objetos OMG³²
- B. Un lenguaje para definir objetos(ODL),cuya sintaxis se basa en un IDL del OMG
- C. Un lenguaje para consulta y actualización de objetos(OQL)basados en aspectos de SQL³³, con las variaciones requeridas para soportar las capacidades de un lenguaje orientado a objetos
- D. Un enlace C++ con Smalltalk en el cual se define como escribir programas portables, que manipulen objetos persistentes, no necesariamente pueden ser estos lenguajes quizá sea Java.

El ODMG-93 agrega conceptos básicos

- Objeto
- Tipo
- Operación

³² OMG Object Management Group

³³ SQL Structure Query Language Lenguaje Estructurado de Consulta

- Propiedad
- Del MO de ODMG los propios para el manejo de bases de datos, tales como:
- Persistencia
- Relaciones
- Transacciones.

Elementos del ODMG-3

Objetos

Como se ha mencionado, cada entidad del mundo real es MODELADO como un OBJETO el cual se asocia a un ESTADO y un COMPORTAMIENTO

- Estado, valores de las propiedades.
- Comportamiento, operaciones que actúan sobre el estado del objeto
- Los objetos se caracterizan por tipos.
- Una base de datos almacena objetos.
- Con la posibilidad de ser compartidos por múltiples aplicaciones y usuarios.

El ESTADO de los objetos puede modificarse durante su vida o bien puede permanecer inmutable.

En el primer término se hablan de objetos.

- └ MUTABLES
- OBJETOS

En segundo término .

- └ OBJETOS INMUTABLES
- LITERALES

Otra forma de clasificar a los elementos del MO es mediante su estructura

Objetos Atómicos(definidos por el usuario) o bien de tipo de colección

Digamos que una colección tiene un número arbitrario de elementos todos del mismo tipo. ejemplo:

- Lista de empleados
- └ Conjunto de discos

Las colecciones pueden ser

- Arreglos
- Listas
- Colecciones de elementos

En las listas se incluyen las cadenas de bits y la de los caracteres.
Los objetos estructurados pueden componerse libremente, ejemplo:

- Conjunto de listas
- Listas de listas
- Arreglos

De tal manera que este objeto podemos definirlo como un DICCIONARIO, como un conjunto ordenado de estructuras cada una con dos campos.

- CLAVE
- IDENTIFICADOR

Las literales pueden ser

- ATOMICAS
- COLECCIONES
- ESTRUCTURADAS

Una literal atómica puede ser

- Número entero o real
- Carácter
- Octeto
- Decimal
- Booleano
- Cadena
- Enumeración

Las colecciones inmutables pueden ser

- Conjuntos
- Sacos
- Arreglos
- Listas

En las literales estructuradas se encuentran

- Fecha
- Hora
- Marca de tiempo
- Intervalo definidos como la especificación del ANSI SQL.

Identidad

Una característica básica de cualquier objeto es una identidad, la cual es inmutable e independiente de sus propiedades y comportamiento ejemplo:

La identidad de un AFILIADO no cambia aunque cambie su sueldo, dirección u ocupación

Todos los elementos del MO tienen identidad, cuya representación es distinta en las literales y en los objetos. La identidad de toda literal está dada por la sucesión de bits que codifican su valor. La identidad de un objeto se implementa por medio de un identificador único llamado OBJECT IDENTIFIER(OID) que es generado por el sistema. En OMG no se especifica ni el tamaño ni la estructura interna de un OID.

El concepto de identidad es opuesto al de CLAVE del modelo relacional. De manera que una clave es única dentro de una relación de tanto que el OID es único en la base de datos

Mediante OID's es posible construir objetos complejos debido a que pueden identificar los diversos componentes del objeto complejo mediante los OID los objetos se pueden compartir permitiendo distinguir entre dos objetos que tiene como componente objetos que son iguales y dos objetos que tiene el mismo objeto como componente

Además del OID asignado por el sistema, el programador puede proveer de un nombre descriptivo a cada objeto que utilice en sus programas.

El sistema de bases de datos orientado a objetos proporciona una función para mapear un nombre de objeto a su OID, tal nombre debe de ser único en la base de datos

Estado

El objeto se define por el valor de las propiedades del mismo, éstas pueden ser atributos del objeto en si o relaciones entre dicho objeto y uno o varios objetos distintos ejemplo

Una persona puede tener como propiedades el atributo ESTATURA y una relación de HERMANDAD con otras personas

El valor de cualquier atributo es una literal (conjunto de literales), pueden accederse por un par de funciones predefinidas. Los atributos no pueden propiedades ni ser subtipos,

Una relación³⁴ es un tipo de propiedades definida entre dos objetos, se permiten relaciones binarias con funcionalidad

- uno-uno
- uno-varios
- varios-uno

¹ Relationship

Un profesor imparte cursos y cada curso es impartido por un profesor
Las relaciones mantienen la integridad, si se elimina un objeto de una relación cualquier intento posterior de partición en ella produce una excepción. Las relaciones de las instancias OID's, se identifican por los objetos que participan en ellas

Comportamiento

El comportamiento potencial de un objeto se define por el conjunto de operaciones que pueden ejecutarse sobre él.

Las operaciones se definen sobre un solo tipo. Los nombres de operaciones pueden repetirse siempre y cuando ellas estén definidas sobre diferentes tipos, al invocar una operación sobrecargada se elige aquella que concuerde con el tipo del objeto proporcionado como primer argumento de la llamada. Debido a la jerarquía de tipos el argumento puede tener varios tipos, en este caso se selecciona la operación definida sobre el tipo más específico.

Tiempo de Vida

El tiempo de vida de un objeto se especifica en la creación del mismo y no pueden modificarse en ningún momento. El modelo de objetos contempla dos tiempos de vida para objetos.

- Transiente Significa que el objeto será almacenado por el lenguaje de programación en tiempo de ejecución, generalmente se declara en el encabezado de un procedimiento y por tanto vive sólo durante la ejecución del mismo
- Persistente Significa que el objeto será manejado por el sistema de bases de datos y que existirá aún después de que el procedimiento o proceso que lo creó haya terminado

El tiempo de vida de un objeto es independiente de su tipo, por tanto un tipo puede tener algunas instancias persistentes y otras transientes y ambas pueden manipularse usando las mismas operaciones.

Tipos y Clases

Los objetos pueden categorizarse en tipos. Un tipo define el conjunto de propiedades a través de las cuales sus instancias pueden interrogar y en algunos casos manipular directamente su estado. Todos los objetos se crean como instancias de su tipo y éste no se puede cambiar. El conjunto de todas las instancias de un tipo se llaman extensión.

Todo tipo tiene una interfaz dentro de una base de datos y una o mas implementaciones aunque sólo una implementación puede usarse en un programa particular. En la interfaz se define el estado visible y el comportamiento de las instancias del tipo. El estado se define mediante firmas para atributos(nombre y

tipo) y para relaciones (el tipo del otro objeto involucrado en la relación y el nombre del enlace). El comportamiento se define como un conjunto de firmas de operaciones. La implementación de una operación se conoce como método

Los atributos definen estados abstractos, por ello están dentro de la definición de la interfaz y no en la implementación, sin embargo, esto no implica que un atributo tenga que estar implementado como un campo dentro de una estructura de datos, ejemplo:

Atributo EDAD de un tipo de PERSONA puede implementarse como un método que la calcula a partir del atributo FECHA_NACIMIENTO y de la fecha actual, en lugar de almacenarlo como propiedad del tipo PERSONA

La implementación de un tipo consta de una representación (conjunto de estructuras de datos) y de varios métodos (procedimientos), uno por cada operación definida en la especificación. Los métodos implementan el comportamiento especificado por su operación asociada modificando la representación del objeto o invocando las operaciones definidas sobre objetos relacionados. Puede haber método y estructuras de datos que no tengan contraparte en la interfaz por ser locales y que sólo son empleados para la implementación de los métodos de la interfaz

Una clase es la combinación de la especificación de la interfaz de un tipo y una de las implementaciones del mismo. Un objeto individual es instancia de una clase se usa para implementar todas las instancias del tipo. El modelo de objetos no especifica si un objeto debe conservar la implementación elegida o si puede cambiarla.

El modelo de objetos es fuertemente tipado:

Cada objeto o literal tiene tipo y cada operación requiere el tipo de sus operandos. Dos objetos o literales tienen el mismo tipo si y sólo si son instancias del mismo tipo.

Los tipos se organizan en jerarquías, el subtipado es una relación entre tipos que define las reglas para que objetos de un tipo sean aceptados en el contexto de otro tipo.

Un objeto de tipo TS (subtipo de T) puede asignarse a un objeto del tipo T, pero el caso contrario no es posible. Las instancias de un subtipo cuentan con las propiedades (ESTADO y COMPORTAMIENTO) del supertipo así como propiedades únicas del subtipo debidas a su naturaleza especializada. Los subtipos pueden tener varios tipos padres, con la implicación que una instancia de un tipo S también lo es de todos sus supertipos. Ejemplo.

Ayudante de profesor puede verse como un subtipo de EMPLEADO y de ALUMNO.

Algunos tipos son directamente instanciables, otros no. los denominados tipos abstractos definen las características heredadas por subtipos sin definir una

implementación, por tanto no pueden ser directamente instanciables. Un subtipo insaciable de un tipo abstracto debe definir una implementación que soporte cada característica heredada de supertipos abstractos.

Cada base de datos es instancia del tipo DATABASE que tiene operaciones para abrir una base de datos, cerrarla, determinar si está vacía o no, y para buscar algún objeto, además de soportar operaciones diseñadas por el administrador de la base de datos.

Transacciones

Los programas que usan datos persistentes se organizan en transacciones definidas como unidades:

- ATOMICAS
- CONSISTENTES
- INTEGRAS
- PERSISTENTES

Todos los accesos, creaciones, modificaciones y supresiones de los objetos persistentes deben hacerse dentro de una transacción.

Para el control de concurrencias se usa el enfoque estándar basado en candados(locks), que pueden adquirirse sobre objetos particulares, y el modelo pesimista en el cual para acceder a un objeto es necesario adquirir un candado que puede ser para lectura, o bien para escritura según sea la operación que se desea realizar con el objeto. Las lecturas sobre un objeto no entran en conflicto con otras lecturas del mismo objeto, sin embargo, los escritores de un objeto entran en conflicto tanto con los lectores como los escritores de ese mismo objeto. Si hay conflicto, el que tiene el candado concedido continúa trabajando y la transacción que lo solicita tiene que esperar hasta que el otro termine y libere el candado.

CONCLUSIONES

- De acuerdo al punto 2.1 las características que comparte el modelo entidad/relación se perciben claramente establecidas ya que fueron justificadas científicamente bajo el álgebra relacional, que hace más de un siglo que se descubrió y que vino a apoyar la estructura y la creación de una base de datos
- En el caso de la orientación a objetos con otros modelos del punto 2.3 que no cuenta con una justificación científica que ampare los conceptos, que sin embargo se ha utilizado pero debido a su complejidad no ha sido establecida en las consultorías como a priori del modelo entidad/relación
- Aunque difiere en algunos puntos y hasta los supera de forma funcional con respecto a los modelos semántico, jerárquico y de red, el modelo orientado a objetos, no es lo suficientemente completo para establecerse de manera inmediata.
- Los estudios realizados a la orientación a objetos ha llevado a establecer ideas más específicas que en la realidad se ven muy relacionados con el modelo entidad/relación
- El modelo orientado a objetos que se estudia en el punto 2.6 no difiere con el MODELO como unidad, descrita en el capítulo anterior, difiere en los conceptos propios de la orientación a objetos, los cuales forzosamente tienen que ser adecuados en el modelo
- Con la utilización del modelo ODMG3 visto en el punto 2.8, las bases de datos serán en un futuro cause de transformaciones sociales que hoy en nuestros días vemos con el Internet, el cual con el poder de los nuevos procesadores y el aumento de los discos duros o periféricos podremos contar con bases de datos inmensas
- Con respecto al punto 2.3.1 el modelo de datos semántico figura 2.9 contó con 10 características, cada una con un valor, de las cuales obtuvo 6, por lo cual aun siendo menos expresivo con menos versiones, no contar con objetos compuestos, y con abstracción es más completo en relación al modelo orientado a objetos
- En el punto 2.3.2 en la figura 2.10 el modelo de red y jerárquico contó con 5 puntos relevantes en comparación con el modelo orientado a objetos, este modelo también resultó cumplir con 4 puntos, y aunque la diferencia es de solo 1 con respecto al modelo orientado a objetos, el modelo de red y jerárquico es más completo
- En comparación con el modelo relacional en el punto 2.3.3 el modelo orientado a objetos resultó más completo ya que obtuvo 6 de los 6 características
- Por lo tanto de los 21 puntos de comparación el modelo orientado a objetos obtuvo 13 puntos lo que equivale a un 60% del total de características, y aunque este porcentaje no deja claro si el modelo orientado a objetos se puede retomar para el diseño de aplicaciones, si puede ser sustituido en relación al modelo entidad-relación ya que los supero en sus 6 puntos con respecto a este
- El modelo ODMG-3 fue diseñado para estandarizar el MODELO ORIENTADO A OBJETOS y establecer una funcionalidad en los manejadores y sistemas operativos así como lenguajes de programación, lo descrito en el punto 2.8.2 pero no ha sido establecido debido a que existen diversas agrupaciones que se oponen al cambio o que luchan por conseguir la función más estrecha a la orientación a objetos. y como la suma de capitales a sido enorme los grandes consorcios se ven obligados a mantenerse como líderes del mercado y por consiguiente cada periodo de tiempo sacan al mercado un nuevo manejador o herramienta enfocada a la orientación a objetos.

3.EJEMPLO DE UNA BASE DE DATOS UTILIZANDO EL MODELO ORIENTADO A OBJETOS.

3.1. ESCENARIOS Y LA ORIENTACIÓN A OBJETOS

Estableciendo diferentes escenarios dentro de la orientación a objetos podemos decir que no ha existido las bases que enriquezcan dicha tecnología en México, en realidad no contamos con bibliografía mexicana que refleje el estudio de la orientación a objetos

De esta manera ¿Cual será el futuro de la orientación a objetos mientras que los científicos, administradores, maestros, planificadores no utilicen estos conceptos y por consiguiente no se desarrollen modelos de datos que den pauta para la creación de nuevos modelos o sustitución de otros?

Una cosa que se debería de dividir son todas las ramas de la orientación a objetos y que no solamente tienen aplicaciones computacionales, en la vida pública. la planeación de una ciudad. los problemas del agua para abastecer a toda una comunidad. en la reutilización de luz solar que disminuiría considerablemente el gasto público, que tan importante sería contar con una ingeniería de la orientación a objetos aplicada a la planeación. cause de muchos reingenierías aplicadas a un sistema tradicionalista capaz de competir con países desarrollados.

Para lograr esto será esencial contar con una orientación a objetos que se desarrolle no solamente con conceptos abstractos y limitados a la computadora, si no a todas las ramas interdisciplinarias que pueda manejar

Es por esta razón que a pesar de que no exista un modelo de datos para las bases de datos orientado a objetos comprobado científicamente, cada administrador o programador tiene la obligación de dar una aportación a esta magnifica rama de las ciencia computacional.

Pero que pasará si en la realidad las demostraciones de operaciones abstractas de la orientación a objetos son justificadas científicamente y la tecnología mecánica, el hardware, tiene la capacidad de almacenar o de respaldar conceptos de la orientación a objetos, en la actualidad son demasiadas las universidades, compañías que se han encargado de desarrollar sus propios modelos y sus propias máquinas, así que la competencia entre los monopolios de computadoras ha aumentado para bien, en este aspecto únicamente, ya no pasará mas de cinco años para establecer la orientación a objetos como una realidad y si a esto se le agrega la influencia que ha tenido la orientación a objetos a la inteligencia artificial, o sobre las grandes cantidades de información que requieren respuestas en tiempo real, en los mapas de localización exacta en donde se necesitan millones de coordenadas para establecer un punto o mejor aun la creación de procesos de simulación que requieren una infinidad de datos para aplicar millones de escenarios en algunos minutos.

Se espera que dentro de poco exista un modelo establecido, de qué dependerá, de los avances que consigo traiga la tecnología, de instrumentos portables y de menor precio de acuerdos internacionales para implementar un modelo, que en un caso mas drástico se espera que la orientación a objetos se verá reflejada como una herramienta metodológica y practica de la planeación o del análisis, y no únicamente como un concepto computacional.

Se tendrá que realizar una reingeniería de nuestro mismo comportamiento ya que será no solamente un cambio de máquinas o computadoras será la era de la inteligencia, de los sistemas expertos, de las bases de datos orientados a objetos de la culminación de programas estructurados, de la revalorización de lo positivo por lo negativo, mientras suceda este cambio la comunidad estará susceptible y puede que en diferentes sociedades se adapte fácilmente, por lo que ayude a soluciones para la medicina, para la ciencia, para la biología, para el avance puramente científico.

3.2. MODELO CENTRAL

Características

- ☐ INTEGRIDAD
- ☐ UNIDAD DE LOS VALORES DE UN ATRIBUTO
- ☐ POSIBILIDAD DEL VALOR NULO PARA UN ATRIBUTO
- ☐ RANGO DE VALORES QUE UN ATRIBUTO PUEDE ASUMIR
- ☐ RELACIONES SEMANTICAS

Considerando un MODELO CENTRAL podemos decir que existen otros modelos como son:

- ☐ MODELO GEMSTONE(1989)
- ☐ O2(1990)
- ☐ IRIS(1989)
- ☐ ODMG(1990)

Y sus componentes varían entre los recursos de sus modelos, lenguajes de acceso y manipulación.

Objeto

Objeto: cada entidad del mundo real se representa por objetos, el cual está asociado a un estado y a un comportamiento.

- Estado: se representa por los valores de los atributos del objeto
- Comportamiento: se define por los métodos que actúan sobre el estado del objeto cuando se invocan las operaciones correspondientes.

En general, cada objeto está asociado con:

- 1 Un conjunto de variables que contienen los datos del objeto (las variables se corresponden con los atributos del modelo E-R).
- 2 Un conjunto de mensajes a los que responde, cada mensaje puede no tener parámetros o tener uno o varios.
- 3 Un conjunto de métodos, cada uno de los cuales es código que implementa un mensaje, el método devuelve un valor como respuesta al mensaje. (véase Fig 3.3)

La única interfaz externa presentada por cada objeto es el conjunto de mensajes a los que responde, por lo tanto resulta posible modificar las definiciones de los métodos y de las variables sin afectar al resto del sistema, que es una ventaja de la PROGRAMACION ORIENTADA A OBJETOS "Reutilización del sistema"

Métodos

La definición de un METODO consiste en dos componentes: encabezamiento y cuerpo. El encabezamiento especifica el nombre del método, los nombres y clases de sus argumentos y la clase de resultado (si se retorna uno)³⁵, mientras que el cuerpo representa la implementación del método y consiste en un conjunto de instrucciones expresadas en algún lenguaje de programación.

Cada objeto puede tener diferentes métodos, los cuales se clasifican como:

- 1 SOLO LECTURA: estos métodos no afectan el valor de las variables de los objetos.
- 2 ACTUALIZACION: estos métodos modifican el significado de los objetos.

Los atributos del modelo E-R representados dentro del modelo orientado a objetos pueden ser mensajes de SOLO LECTURA. Por ejemplo, tenemos la tabla PRTF (Afiliados), la cual tiene un campo denominado Fecha de Nacimiento FECHNAC, el cual claramente podemos identificar que nos da la fecha de nacimiento del empleado. Si lo manejaremos como objetos podemos definir que este FECHNAC sea un método el cual calculará la fecha de nacimiento restando la fecha actual

³⁵ BERTINO, Martín. Sistema de Base de Datos Orientado a Objetos. Definición.

menos la fecha de nacimiento, por lo tanto quizá en otro sistema se quisiera calcular el promedio de edad de los empleados de un año a la fecha ,no se tendrá que agregar un campo que describa la edad de 30 a 35 años, en la base de datos , se tendría que cambiar la estructura de la tabla se modificarían los programas y el tiempo de proceso aumentaría ya que el tamaño de esta tabla crece continuamente. De lo contrario utilizando el concepto de orientación a objetos únicamente el METODO se tendría que actualizar, así que el objeto seguiría siendo el mismo con mas mensajes uno puede ser FECHANAC y otro FECHPROMEDAD

Mensaje

El término mensaje en orientación a objetos es un elemento que se utiliza para mandar llamar métodos entre los objetos y la ejecución del método correspondiente y no significa que se manden mensajes físicos en redes informáticas

Identidad

Cada objeto se identifica por un simple IDO(Identificador de Objetos),la identidad de un objeto tiene una existencia independiente de los valores de los atributos del objetos, por ejemplo:

Definamos el **Objeto:**
PERSONA
Atributos:
Alto,
Bajo ,
Gordo,
Flaco
Identidad:
Flor,
Juan,
Alex

Se podría confundir la identidad de un objeto como un atributo del objeto pero es primordial definir una IDENTIDAD DEL OBJETO para reconocer sus atributos

□ Objetos y Valores

En algunos modelos en los cuales se permiten objetos y valores. y donde no todas las entidades se representan como objetos.

Un valor es autoidentificable y no tiene ningún IDO asociado a él

□ Entidades Primitivas

Ejemplo los enteros o caracteres, se representan por valores.

□ Entidades No Primitivas

Ejemplo objetos.

□ Valores Complejos

Los cuales no pueden ser compartidos por otros objetos: ejemplo conjuntos, como componentes de otros objetos, pero que no se utilizan como entidades independientes, fechas.

□ Diferencias en comparación con el concepto de clave

Este concepto que proviene del MODELO RELACIONAL es el concepto clave, un atributo o un conjunto de atributos cuyos valores se identifican unívocamente solo y solo una vez cada tupla, como el mencionado CURP (Clave Unica de Registro Poblacional) que identificará a una y solo una persona durante toda su vida

Para empezar a definir todo el concepto de una base de datos orientada a objetos ,al lo que se pretende llegar tenemos una tabla, entidad de la base de datos del Sistema de Ahorro de Fondo para el Retiro, administrada por PROCESAR S A. de C. V la cual esta conformada por una infinidad de tablas que a su vez forman un base de datos, para ser mas específicos tomemos una de estas tablas y llevémosla para describir algunos de sus campos(véase fig 3.1)

Tabla PRTF
Nombre
Apellido Paterno
Apellido Materno
Dirección
NSS
RFC
Fecha deProceso
Clave de Afore
Identidad del Proceso

Fig. 3.1 Tabla PRTF del SAR.

En esta tabla se concentran, guardan, almacenan todos los empleados que están registrados como AFILIADOS y que cuentan con una AFORE, esta tabla se alimenta día con día ya que diario se actualiza con nuevos registros de los empleados que se les asigna una AFORE

Ahora bien se puede encapsular todo el código del proceso que calcula las aportaciones individuales ,así como sus intereses que genera.

Por lo tanto al hacer referencia al objeto de EMPLEADOS AFILIADOS responderá al mensaje de APORTACIONES ,de tal manera que la interface que se muestra en EMPLEADOS AFILIADOS siempre será la misma ,con la funcionalidad de que cada uno se actualizará de manera diferente.

Siendo conceptualmente rigoristas en el MODELO ORIENTADO A OBJETOS los atributos de las entidades serán como variables y un par de mensajes del objeto correspondiente

La variable se utiliza para guardar el valor del atributo

Uno de los mensajes se utiliza para leer el valor del atributo

Otro mensaje se utiliza para actualizar el valor .

Por ejemplo la tabla PRTF cuenta con un campo DIRECCION el cual puede representarse :

Una variable DIRECCION

Un mensaje OBTENER-DIRECCION, lo que responderá a la dirección de cada empleado.

Un mensaje DEFINIR-DIRECCION ,el cual necesita de una nueva variable DIRECCION en el caso que se tenga que ser actualizado

Clases de Objetos

Dentro de la base de datos hay muchos objetos similares, *similar* se entiende que responden a los mismos mensajes , utilizan los mismos métodos y tienen variables del mismo nombre y del mismo tipo, por lo que sería una pérdida de tiempo, análisis y horas/hombre definir por separado cada uno de estos objetos Por lo tanto los objetos parecidos se agrupan en CLASES, que al formar un grupo que representa una clase a cada uno de estos se le define como *ejemplar de su clase* , todos los objetos de una clase comparten una definición en común, pese a que se diferencien en los valores asignados a las variables.

Para representar mejor el concepto de clase e identificarlo de el modelo E-R con el modelo orientado a objetos una clase es una entidad ,algunos ejemplos de clase en la base de datos son PRTF(Afiliados), PRTS(Trasposos), PCRQ(Rechazados por Detalle), PCET (Descripción de Afores), PLDM(Proceso de Liquidación (véase fig 3 2)

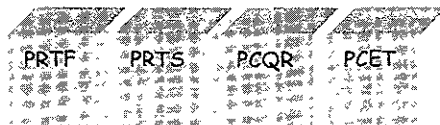


Figura 3.2 Entidades o Clases

A continuación se mostrará en pseudocódigo la clase de EMPLEADOS

```
Class EMPLEADO {
    /* Variables*/
    string Nombre;
    string Apellido Paterno;
    string Apellido Materno;
    string Dirección;
    string NSS;
    string RFC;
    string Afore;
    string Clave Afore;
    /* Mensajes*/
    integer Aportaciones;
    string Obtener-Dirección;
    string FechPromEdad;
    string Obtener-Nombre;
    string Nueva-Dirección;
```

Obsérvese que de la clase de EMPLEADO se definieron sus variables y sus mensajes.

El concepto de clase es parecido al de tipo abstracto de datos.

En algunos momentos pudiera ser que se llegue a confundir con un objeto, llamado objeto clase pero existen algunas diferencias que se describen a continuación:

- Una variable de tipo conjunto cuyo valor es el conjunto de todos los objetos que son ejemplares de la clase.
- La implementación de un método para el mensaje nuevo, que crea un nuevo ejemplar de la clase.

Herencia

Representar una base de datos orientada a objetos puede llegar a utilizar un número finito de clases, frecuentemente varias clases son parecidas, en la base de datos que estamos haciendo referencia se puede decir que las tablas siempre son mencionadas las CLAVES, que si una identifica la AFORE otra la RECAUDADORA otra la AFORE MAQUILADORA y en cada tabla estos campos se ven reflejados, por lo tanto para definir a la CLAVE AFORE es una especialización de CLAVES y que CLAVE DE AFORE MAQUILADORA es una especialización de CLAVES. Por lo que se puede decir las variables y los métodos específicos de la CLAVE AFORE se asocian a la clase de CLAVE, de igual manera las variables y los métodos específicos de CLAVE AFOREMAQUILADORA se asocian a la clase de CLAVE.

- Existe un conjunto A llamado PERSONAS que a su vez pueden ser AFILIADOS NO AFILIADOS(AM)

3.3. APLICACIÓN

Para complementar los conceptos que integra la tecnología orientada a objetos, aplicado en una base de datos se analizará la BASE DE DATOS que fue desarrollada bajo un modelo entidad-relación que forma parte principal del Sistema de Ahorro para el Retiro, es importante hacer notar que el primordial componente es la base de datos por lo que no se reflejará las descripciones o se cambiarán nombres de tablas y se cuidará que el modelo conceptual no produzca alguna discrepancia entre sus autores.

3.3.1. CARACTERÍSTICAS

El sistema está dividido en varios subsistemas los cuales cuentan con un diagrama de procesos con una serie de programas en COBOL, un sistema operativo TSO que utiliza elementos identificados como JCL's, PROC's, un manejador de base de datos DB2 con un conjunto de tablas relacionadas, especificaciones por programa, por flujo, el cual ha sido desarrollado durante 6 años.

Y para controlar tan inmensa información así como la funcionalidad sea bien vista por los institutos que la integran , el sistema se ha desarrollado bajo reglamentos y especificaciones del gobierno federal y de instituciones involucradas como AFORE, ICEFAS ,CONSAR , INFONAVIT e IMSS.

El sistema esta soportado bajo un ambiente de maiframe ,una HP900 con infinidad de terminales.

Anteriormente se mencionó que el sistema estaba dividido en subsistemas a saber.

- Traspasos
- Recaudación
- Retiros
- Reversos
- Afiliación
- Liquidación

Para el caso de estudio se utilizará el subsistema de Traspasos

3.3.2. SISTEMA DE TRASPASOS

3.3.2.1. CONCEPTO GENERAL

“El nuevo sistema de pensiones establecido por la CONSAR especifica que cada trabajador tiene el derecho de elegir AFORE de su preferencia “(ley del sistema de ahorro para el retiro,art 74) Por lo tanto este podrá hacer el traspaso de sus recursos a la AFORE deseada

3.3.2.2. SISTEMA CONCEPTO

El sistema recibirá procesará y enviara solicitudes de Traspasos que contempla

- 1 Traspasos.
2. Devoluciones de Solicitudes
- 3 Solicitudes
4. Traspasos Complementarios
- 5 Solicitudes no Atendidas
- 6 Traspasos Exitosos.
- 7 Liquidación

Para detallar mas el caso de estudio se enfocará en Recepción de Solicitudes de AFORE-AFORE

3.3.2.3. TRASPASOS AFORE-AFORE

Este traspaso se lleva a cabo cuando un trabajador que ya está afiliado a una AFORE decide cambiarse a otra AFORE que le ofrezca mejores beneficios.

EL proceso cuenta con un flujo(véase fig 3 4)

Cada cuadro representa un JCL, PROC y a su vez una breve descripción de lo que calcula o realiza, que a su vez cuenta con una serie de programas que contienen tablas o entidades que conforman la base de datos.

Cabe señalar que la información que corre a través del flujo llega por medio de archivos secuenciales, archivos de texto , con una longitud , tamaño y un número finito de registros, que al inicializarse el proceso se registran por número de lotes. lo que significa que pueden llegar varios lotes.(véase fig 3.3)

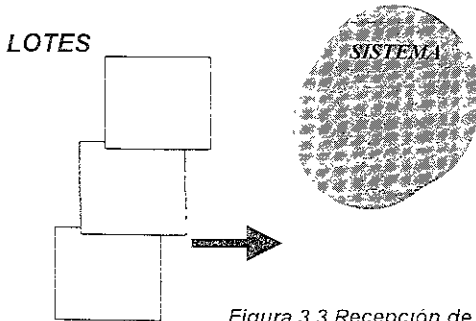


Figura 3.3 Recepción de Lotes AFORE-AFORE

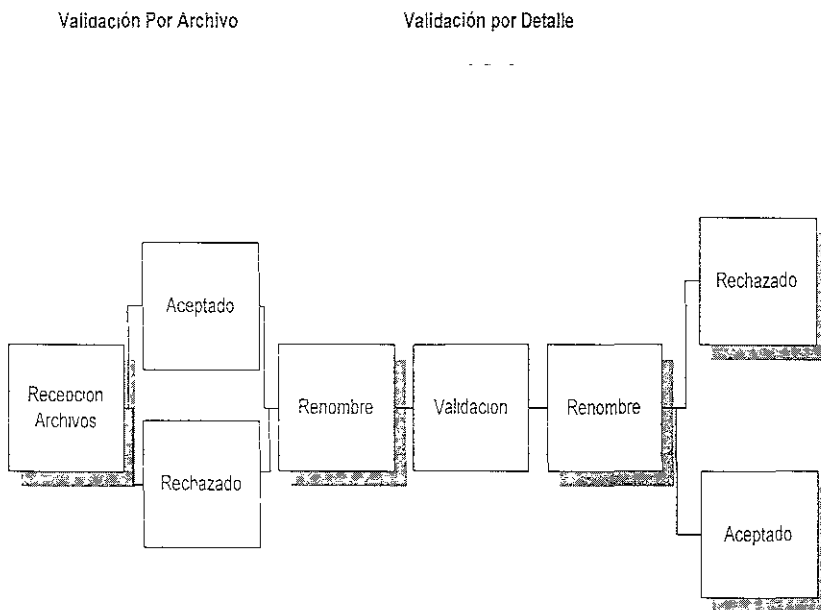


Figura 3.4 Flujo de Solicitudes AFORE-AFORE

La tabla siguiente describe la figura 3 4

Recepción de Archivos	Valida Lotes Recibidos por Afore
Aceptado	Lote Aceptado
Rechazado	Lote Rechazado
Renombre	Genera proceso de validacion de movimientos a detalle
Validacion	Validacion de registros por día
Renombre	Registros Rechazados
Rechazado	Genera proceso de envío de movimientos rechazados
Aceptado	Genera proceso de envío de movimientos aceptados

Teniendo en cuenta las descripciones, definiciones y elementos que conforman la tecnología orientada a objetos y de acuerdo a la figura 3 3 que describe el proceso de AFORE-AFORE estos serían los elementos a crear en los primeros pasos para conformar una base de datos orientada a objetos

El primer elemento que se ha mencionado es el OBJETO, que dentro del sistema llega a ser único por cada EMPLEADO que se almacene

Objeto

Empleado

Por conveniente es necesario asignarle una serie de atributos que lo IDENTIFIQUEN como único, los siguientes son:

Atributo

Nombre
Apellido Paterno
Apellido Materno
Dirección
NSS
CURP
RFC

Para enfocar mas el sistema mencionamos algunos métodos que se utilizan en el proceso de AFORE-AFORE

Método

Verifica fecha de proceso.

Diariamente se reciben OBJETO.EMPLEADO y para ubicarlo se le asigna una fecha ,que significa el día en que se procesa que tiene que ser igual a la fecha del sistema , a la fecha que tiene automáticamente la computadora. Esto quiere decir que si un OBJETO.EMPLEADO llega con una fecha de 25 de noviembre de 1999, ese día se procesó y tuvo que tener la fecha de la computadora 25 de noviembre de 1999.

Rechazo de empleado por fecha.

Si el OBJETO.EMPLEADO su fecha es diferente a la de la computadora, se rechaza

Guarda empleado rechazado por fecha.

El OBJETO.EMPLEADO cuenta con un rechazo para identificarlo se almacena como rechazado

Identifica el proceso a ejecutar.

Este método identifica que proceso de Traspasos va a ejecutar ya sea un Traspasos, Cambio de Afore Devolución , Traspaso Complementario, etc

Identificación de movimiento.

Este método podrá identificar si se trata de una AFORE, ICEFA ,AFORE MAQUILADORA RECAUDADORA

Define un nombre de acuerdo al proceso

Otorga un nombre que coincide con el proceso que se ejecutó

Define un consecutivo

Por cada empleado que se procese asigna un número consecutivo.

Valida el empleado por detalle

En cada OBJETO EMPLEADO se valida los atributos que puede tener cada EMPLEADO OBJETO

Rechazado por detalle

Se le asigna una clave que identifica que rechazo le corresponde por OBJETO.EMPLEADO.

Verifica periodo de proceso

El OBJETO.EMPLEADO cuenta aparte de una fecha con un periodo este periodo es por mes.

Aunque los métodos que son demasiados y no se mencionan completamente, si nos puede dar una idea de que el proceso se identifica de acuerdo a un flujo que métodos a utilizar.

3.4. COMO SELECCIONAR UN MODELO

De acuerdo a los diferentes modelos de datos que pueden ser utilizados para desarrollar una base de datos orientada a objetos, el hecho de que un modelo sea caso de estudio de un administrador DBA, sino de analistas, programadores, líderes y todo el personal que componga el desarrollo de una aplicación y siguiendo, por lo tanto un orden y una metodología, la elección de un MODELO contempla cuatro pasos

3.4.1. Grupo UNO

- La información puede almacenarse utilizando programación orientada a objetos.
- El utilizar programación orientada a objetos permite almacenar información en una entidad de manera segura como también acceder a ella mediante múltiples usuarios al mismo tiempo.

3.4.2. Grupo DOS

- Uno de los objetivos de los administradores de las bases de datos es buscar diferentes caminos para almacenar datos, lo cual lleva a buscar el mejor performance, así como el modelo de datos que se ajuste a las necesidades de la empresa.
- Para esto deben de comparar los diferentes modelos que hay en el mercado como son relacional, objetos, jerárquico, red, funcional

3.4.3. Grupo TRES

- Este grupo hace énfasis en las personas que administran proyectos y se preguntan si la tecnología a utilizar es real, por lo cual, se encargarán de conocer las ventajas y desventajas de la misma, es importante resaltar que existe en el mercado tanto software y hardware, para saber colocar los componentes que tengan más éxito, y sin olvidar en el caso de las bases de datos el elegir el mejor modelo que se ajuste al sistema de cómputo.

3.4.4. Grupo CUATRO

Por último al utilizar un administrador de base de datos (DBMS) para objetos, es importante contar con una guía de selección, así como de sus aplicaciones con las

que cuenta, lo cual esto acompaña a la selección del modelo el cual implicará decisiones técnicas, estrategias de negocios y la selección del equipo.

La evaluación y selección de un modelo lleva entre cuatro y cinco meses

De manera mas detallada³⁶ considere tres fases para seleccionar el MODELO, como también comprender los conceptos básicos de tecnología de objetos, tecnología de DBMS, y el lenguaje de programación a utilizar

La primera fase comprende

- Conceptos básicos que encierra la orientación a objetos.
- Conocer la tecnología de los manejadores de bases de datos(DBMS).
- Elegir un lenguaje de programación que será utilizado para el desarrollo de las aplicaciones

La segunda fase implica.

- Adentrarse de manera mas específica y detallada en el manejador de bases de datos orientado a objetos, a través de procesos que interactuen con las características del producto aplicaciones y la relación entre estas

La tercera fase y última describe

- Los campos para identificar una lista de características críticas para después utilizar en una selección de una lista pequeña de vendedores que se encuentren en el mercado, con capacidades de contar con test de sus productos de manera que esto lleve a la mejor selección de la aplicación.

3.5. LAS BASES

Tan importante para avanzar y seleccionar el modelo es analizar las bases dicho así consistirá en --conocer las bases básicas-- , de qué, de la orientación a objetos, de los manejadores de bases de datos del lenguaje a seleccionar.

La capacidad del grupo de trabajo para tomar decisiones favorables al momento de elegir el modelo serán fuertemente justificadas bajo la preparación total o bajo el conocimiento que tengan de

- Orientación a Objetos
- Manejadores de Bases de Datos
- Lenguajes de Programación

3.6. LA TECNOLOGIA DE OBJETOS Y DBMS

El equipo de trabajo debe de contar con los conocimientos que abarquen la orientación a objetos

³⁶ Object Database Handbook: How to Select, Implement and Use Object Oriented Databases

Debido a este requerimiento es necesario tener al personal con experiencia o en sus defecto personal que haya tomado algunos cursos o bien aficionados a la orientación a objetos.

Por lo tanto la tecnología debe de ser usada, con el suficiente interés para poder medir el impacto que puede tener en la aplicación a desarrollar.

3.7. LA TECNOLOGIA DE DBMS

Otro punto es conocer el administrador de bases de datos(DBMS), saber cual es su funcionamiento y sobre todo porque es tan importante el DBMS con respecto al modelo.

Saber que lenguaje permite la persistencia de objetos de datos (objetos datos) que sean mostrados puedan utilizar en un ambiente multiusuario, lo cual es crítico para la selección de procesos.

3.8. ESCOGER EL LENGUAJE.

En diferentes situaciones al iniciar los procesos de desarrollo de sistemas se toman como primeras decisiones la selección de un lenguaje de programación, estando aún en las primeras etapas de diseño, y no por ser así deberíamos de pensar que se ha tomado la mejor decisión al elegir x o y lenguaje de programación

- El lenguaje de programación es conocido y usado por uno o varias personas que forman el equipo de trabajo, lo cual no significa que sea este el mas conveniente para la aplicación
- Algo importante el lenguaje debería de cumplir con las necesidades y requerimientos, así como características de la aplicación como del ODBMS. Por lo tanto el lenguaje debe dirigirse hacia la selección del manejador de bases de datos orientados a objetos, aunque contrariamente se escoja primero el lenguaje

3.9. MAS ALLA

En esta parte se gasta mas tiempo por la elección del modelo, ya que el estado del modelo se vuelve interactivo

El caso es la relación que toma el modelo con respecto al manejador y sus características, por lo tanto la selección de un manejador debe de llevarse de la mano con respecto al modelo y esto debe de ser claro; se recomienda tener la antítesis de un manejador, su objeto modelo, prototipo e identificar a los vendedores que aporten una idea clara y específica de los productos de bases de datos

El impacto de la aplicación se considera con el proceso interactivo entre las ideas o características del producto con las de la aplicación.

De manera que la percepción ante la aplicación podrá requerir de diferentes cambios.

3.10. CARACTERISTICAS DE UN DBMS

En la actualidad no hay parecido entre los ODBMS, aunque son muy pocas las características entre estos, lo que se hace es buscar cual es el mejor ODBMS o el que mas convenga, así que no sería conveniente manejar un producto por cada aplicación, a razón de que cada aplicación requiere combinaciones de características

Uno de los problemas podría ser que las personas no conocen o mejor dicho no ofrecen totalmente las características de la aplicación lo que provocaría una mala selección

Tomar decisiones basadas en el conocimiento común o en lo que la última industria ha aplicado se llevaría a una informal decisión de mercado.

Por lo tanto existen 13 categorías para elegir y tomar la decisión de permanecer con uno o con otro ODBMS

Las categorías proporcionan el nivel de detalle que incluye cada una serie de características.

Por lo tanto la llave para llevar un proceso exitoso es conseguir que las características de la aplicación se empaten con el mayor número de categorías

Es importante resaltar los puntos críticos de la aplicación primero para poderlos entender, características del producto y como interactúan las necesidades de la aplicación con el producto

Considerar que la combinación de la aplicación con el producto dan como resultado una mejor aplicación, y comprender el problema del negocio para poder direccionarlo.

Como un DBMS puede explotar con respecto a las necesidades de la aplicación.

3.11. LAS CARACTERISTICAS DE INTERACCION

La aplicación no necesita ser una vacuna separada de las características de ODBMS

Las características intercalarán con otras aplicaciones de manera en que el diseñador pueda anticiparlos. La tarea es no caer en la aplicación dentro del producto y que el producto en la aplicación, por lo tanto deben de trabajar en conjunto para proporcionar la función necesaria de la aplicación.

Dos características toman únicos caminos en la aplicación puede producir un desastre o un suceso inesperado, características como.

- Locking Granularity
- Proceso de Queries
- Ejecución de métodos

El uso de clases class-level locking es otro ejemplo de una característica crítica de interacción

Y estas se utilizan de manera dinámica cuando se modifican los datos (data definition) tan pronto como las instancias de los objetos de clases puedan cambiarse.

La cantidad de código en los programas eliminado por que puede ser inmensa. Pensando en las diferentes ramificaciones este paso es esencial.

3.12. DE QUE MANERA COMPRENDER LOS REQUERIMIENTOS DE LAS APLICACIONES

Para entender cada aplicación es necesario contestar o considerar las siguiente preguntas

1. ¿Cuál es el ambiente de cómputo?
2. ¿Qué datos serán distribuidos?
3. ¿La distribución de la información será local o abarcará una zona geográfica mas grande?
4. ¿Cuántos usuarios trabajarán en el proyecto?
5. ¿Cuántos usuarios accederán a la base de datos?
6. ¿Los cambios en los tipos de esquema serán parecido?
7. ¿Cuáles son los datos mas solicitados dentro de los queries?
8. ¿Serán utilizados métodos para la creación de queries?
9. ¿Se accederán datos desde fuentes no objeto como un RDBMS?

Ahora bien el proceso de investigación lleva tiempo, así comprender los requerimientos de la aplicación, analizar y responder cada uno de los anteriores preguntas proporciona ayuda a identificar el punto crítico por lo tanto es conveniente estudiarlas y convertirlas en puntos no críticos

3.13. ETAPA FINAL

Una vez que se estudiaron las características de un DBMS, la interacción de la aplicación, el funcionamiento, el tiempo que se dedicó al desarrollo, al finalizar se podrá seleccionar un modelo en donde se podrá elegir los campos o datos de la aplicación, y se escogerá un DBMS del mercado del software que mas se halla ajustado a la funcionalidad operativa de las bases de datos

El encargado o los encargados de elegir un DBMS deberán intercalar lo suficiente entre la aplicación y las características del DBMS; mientras no se justifique la relación, no se estará listo para determinar el modelo, ni tampoco las rutas críticas.

3.14. CARACTERISTICAS CRITICAS

Dentro de cada aplicación hay productos indispensables (características críticas) si un producto se olvida el análisis podrá dejarlo afuera. Esta lista radicalmente reduce el número de productos que antes se buscaban. Pero es indispensable mantener en mente las características críticas que se aplicarán únicamente dentro de la aplicación. Futuras aplicaciones pueden tener diferentes(características críticas) por lo que cada aplicación es diferente y por consiguiente las rutas críticas son diferentes.

3.15. CREAR UNA LISTA DE VENDEDORES

Sea cual sea el producto ha considerar es conveniente elegir una serie de productos que se involucren con la aplicación, que estudien la aplicación, que se sumerjan, en la opinión de los expertos dos o mas vendedores siempre serán buenos consejeros y tratarán de convencer al cliente con todo una serie elementos que justifiquen la compra del producto, pero solamente uno de ellos verá claramente las necesidades de la aplicación

Para cumplir con las características de la aplicación, se definen tres reglas.

1. Cumplir con los requisitos de la aplicación
2. Haber entendido tanto los requisitos y al equipo de desarrollo
- 3 Establecer estrategias de negocio, mercado financiero

Por consiguiente el joven mercado ODBMS y la desigualdad de productos podrá ser establecido después de muchos años

Por lo tanto escoger un vendedor o un producto dependerá del éxito o fracaso de la empresa así como, si el DBMS se adecuó de manera precisa a las necesidades de la aplicación.

Después de seleccionar los campos de la etapa crítica, la cuál ayudará a seleccionar el DBMS, será necesario establecer un performance de requerimientos, sin esto, podrá ser peligroso.

CONCLUSIONES

- En relación a los escenarios el punto 3 1 se concluye que el estudio de la Orientación a Objetos no ha sido desarrollada en comparación con otros países, esto refleja el poco interés y la mala planeación para que en las universidades dedicadas a la investigación de la ciencia de la computación no diseñen planes de estudio que fortalezcan la idea purista de utilizar la tecnología de la orientación a objetos
- Siguiendo este mismo punto, mientras no se difunda la investigación del modelo orientado a objetos en una base de datos, no sólo en las universidades si no en las consultorías no se podrá establecer un estándar del modelo de datos
- De acuerdo al punto 3 2 que describe las características de la orientación a objetos en relación mas a las bases de datos, mientras no se establezca un criterio que no implique mas modificaciones a las teorías no se podrá establecer a corto plazo un modelo de datos orientados a objetos
- La idea básica de la orientación a objetos como se describe en el punto 3 3 promueve de manera eficiente la reutilización de código por lo cual disminuye el número de líneas de programas y establece una planeación de diseño y desarrollo con menor tiempo
- La idea de establecer un modelo de datos orientados a objetos bajo un plan de selección como se describe en el punto 3 4 da la pauta para no solo elegir un modelo de datos sino también de conformar un grupo de trabajo que incluya analistas, programadores, líderes que a su vez establezcan un plan de trabajo de cinco a seis meses
- La planeación estrategica que establece conceptos y teorías de diferentes autores dedicados a la investigación podrá ser empleada si en verdad los consorcios de las bases de datos como son ORACLE INFORMIX, IBM, SYSDATABASE toman en serio esta ideología y no se preocupan por establecer cada uno su propio modelo

CONCLUSIONES FINALES

La creación de un modelo de datos orientado a objetos es adoptado por los fabricantes de software por ejemplo O2 ODMG, Versant, entre otros con base a su complejidad y fundamentos teóricos que justifiquen la creación del mismo, pero también puede formar parte de una aferrada lucha de monopolios computacionales que acaparan a los genios de los sistemas para justificar el modelo del producto que sacan al mercado

Es cierto que la programación orientada a objetos es un concepto de moda y que pese a los programadores de no difundir este paradigma desde 1967 con la creación del lenguaje SIMULA, en estos momentos hablaríamos más de inteligencia artificial o realidad virtual que de identificar un modelo de datos orientado a objetos, la no-identificación de los conceptos abstractos de la orientación a objetos y la utilización de bases de datos orientada a objetos no fue a tiempo o quizá estaba fuera de alcances, como recursos más potentes de computadoras o lenguajes de programación. así como la ideología misma, no nos podemos desprender del escalón que correspondía al modelo entidad-relación elaborado bajo un álgebra relacional descubierta hace más de un siglo y que actualmente, enormes instituciones como PEMEX, IFE y PROCESAR cuentan con bases de datos (very large) bajo un modelo entidad-relación, que a su debido tiempo serán reemplazados por nuevos modelos.

Como todo ciclo de vida los modelos entidad-relación están en un proceso de transición, algunos modelos mantendrán la misma conceptualización otros elaborarán alternativas con orientación a objetos y muchos más brincarán el escalón hacia la orientación a objetos, mientras suceda eso, el momento preciso de seleccionar un modelo de datos orientado a objetos ha llegado, analizando de manera conceptual y reuniendo los elementos más completos de la orientación a objetos pudimos dar al lector el modelo de datos que más se acerca a sus necesidades, por su complejidad, elementos que en unos y otros eran más marcados

y no contando con una experiencia de años atrás a diferencia de otros elementos de software, que ya han sido vistos de manera comercial, si no con base a la gran aportación científica y tecnológica que sus desarrolladores han tenido a lo largo de veinte años, por lo tanto el miedo debe de desaparecer y los programadores, analistas, líderes de proyecto, administradores de redes, bases de datos y sistemas operativos, así como los portadores de los recursos monetarios, tendrán que elegir un modelo de datos orientado a objetos, las justificaciones son varias:

1. El constante crecimiento en volumen de información en sus bases de datos
2. La creación de lenguajes de programación orientados a objetos, reutilizando código
3. La venta o renta de equipos poderosos, no muy caros, para el almacenamiento de la información.
4. La rapidez para acceder a la información, respetando las reglas generales de las bases de datos como integridad, seguridad, fiabilidad, consistencia
5. El desarrollo del mercado por estandarizar normas, parecidas a las de ISO 9000 igualmente establecerías al modelo de datos orientado a objetos.

Estas y muchas mas cambios tendrán que ser evaluadas por los consumidores de software. solo falta esperar que el fantasma de la orientación a objetos se enciave dentro de una organización y de sus desarrolladores.

Por otro lado en donde la planeación juega un papel de suma importancia, habiendo puntualizado en el capítulo dos la comparación de modelos con respecto al modelo orientado a objetos, para los diseñadores que tendrán que tomar la decisión de elegir un modelo para la creación de una aplicación, es importante recalcar la formación de un grupo de trabajo, el cual tenga conocimiento de la orientación a objetos, con esa primera parte, estimar el tiempo de selección, lo que formará un plan de trabajo mas consolidado, no sin olvidar, definir las características de la aplicación, el manejador de bases de datos que es parte del software lenguaje de programación, entrevistas con los vendedores, pruebas de manejadores y por último y lo mas importante considerar las rutas críticas por si algo falla así que de manera general se establecen subsistemas que vienen siendo puntos relevantes de una planeación, por lo que se reconoce que no solamente se requieren técnicas de programación con conceptos abstractos si no también establecer la planeación como una metodología que se menciona en el último capítulo, con relación a la selección de un modelo de datos orientados a objetos

ESCENARIOS

Para elaborar escenarios es primordial contar con una recopilación de información, por ejemplo artículos, revistas, libros y todo lo que pueda desprender ideas referentes al tema que se desea desarrollar, y entre mas se conozca los elementos que puedan estar involucrados, el factor sorpresa disminuirá y se establecerán con objetividad los escenarios, por lo tanto la creación de uno a tres escenarios, representa la forma correcta para ubicarnos en un estado normal, pesimista u optimista

La elaboración de escenarios aunque es complejo en forma, nos sirven para tener presente con mayor lucidez y con una planeación de estados que nos den el resultado que sea deseado o se acerca a la realidad, de preferencia deseamos estar en un estado normal con una retrospectiva hacia el futuro y hacia un escenario optimista, pero no siempre es así debemos tener presente que aunque el factor deseo y optimismo, como la planeación y el trabajo constante, no siempre nos ubicará en el escenario normal, cabe señalar que solamente se habla de estados en los que una investigación como lo es la tecnología orientada a objetos para rediseñar una base de datos entidad relación, puede observarse desde un punto de vista sin escepticismo y sin carácter de incertidumbre, por lo tanto es necesario definir cuales son las variables que son importantes y que en cada escenario sufren modificaciones, para enfocarnos mas al tema al cual se ha desarrollado una serie de escenarios, las variables que describen los escenarios

De forma cualitativas son:

- **Educación:** la cual implica investigación, publicación de libros y artículos, conferencias, planes de estudio, creación de nuevas universidades o departamentos enfocados a la orientación a objetos
- **Modelo Orientado a Objetos** partimos de que tenemos ya las bases para crear un modelo, bases científicas que se han adoptado a las diferentes bases de

datos, como jerárquico, entidad-relación, red, funcional, semántico, por lo cual estamos en espera de que se estandarice al modelo orientado a objetos.

- **Bases de Datos:** como tal siempre tendrá la misma funcionalidad, la de almacenar información de tal forma que para el usuario, para aquel que necesite de datos pueda interpretarlos y tenerlos de una forma fácil y segura, pero no así su forma operativa la cual tendrá que evolucionar debido al cambio de los manejadores, desarrollados por compañías de software como IBM, ORACLE, INFORMIX, SYBASE, Microsoft entre otros
- **Universidades o instituciones** que son fuentes básicas para la creación de empleados con un alto grado de desarrollo e iniciativa para generar nuevas investigaciones, para prosperar en el ámbito computacional, y generar exportaciones y desarrollo de software.
- **Estandarización**, la cual ha sido un problema que hasta hoy no se ve resuelto, por el número de empresas que se dedican al desarrollo de bases de datos y a la creación de su propio modelo de datos orientadas a objetos

De forma cuantitativa

- **El número de Universidades**, de Instituciones en donde se concentre y se cree planes de estudio de forma responsable para aumentar el nivel del estudio, no solo de la orientación a objetos sino de la misma computación
- **Inversión**, como tal es siempre elemento de importancia, la inversión como primera fase, consiste en la aportación de capital el cual es otorgado por empresas privadas o públicas para fortalecer la investigación y la educación
- **Tiempo**, el cual lleva una investigación, la recopilación de información, la comparación de un modelo, la simple planeación al elaborar un plan de estudios.

Por lo tanto, de manera que podamos ser claros y ubicarnos en un escenario ya habiendo definido las variables mas representativas, empecemos definiendo el escenario normal:

Los desarrolladores de software enfocados a los manejadores de bases de datos como ORACLE, INFORMIX, IBM elaboran sus bases de datos, basándose en la creación de un MODELO DE DATOS el cual representa o refleja la integridad y la funcionalidad de la empresa a la cual se le elabora un sistema de cómputo, en el que se analiza desde el sistema operativo que maneja, la cantidad de información con la que cuenta la cantidad de inversión con la que pueda mantener el proyecto, si es o no necesario la creación de una base de datos y o mejor un manejador de archivos, lo que es importante hacer notar es que el manejo de una base de datos debe de ser representado por un modelo, Modelo de Datos que utilice la orientación a objetos, que los desarrolladores, programadores, analistas, lideres de proyecto conozcan la orientación a objetos para aumentar el factor de la estandarización, por lo tanto aumentar la experiencia e incrementar la oportunidad de establecer planes de estudio mas completos, y así crear una nueva forma de pensar para el desarrollo de software

Es importante que en cada proyecto de software, que se desarrolle bajo la orientación a objetos, lleve como máximo de tres a cinco años el desarrollo del sistema, así en universidades como la UNAM, IPN, TEC, ITAM, entre otras, en las áreas enfocadas a la computación, se establezcan planes de estudio en donde la utilización de la orientación a objetos sea tomada como una clase más.

Ahora para un escenario optimista en el que siga reflejándose la variable de la educación, y de la propia investigación, sobre clonación, redes neuronales, realidad virtual, inteligencia artificial, y la creación de bases de datos en donde será inimaginable la cantidad de información, siempre será necesario el manejo de un MODELO un modelo de datos, que este estandarizado y comprobado bajo el método científico,

El último escenario, el pesimista, y que también se ve reflejado en la educación, podría ser que en las universidades de México no se incremente la oportunidad de investigación en dos años, para desarrollar diseños orientados a objetos, y que a pesar de los esfuerzos de los investigadores por estandarizar un modelo, éste no cumpla con el método científico y por lo tanto no se pueda lograr la estandarización adecuada

Al desarrollar bases de datos o cualquier sistema de cómputo no se utilice desde su etapa inicial las metodologías de análisis, diseño e implementación de la orientación a objetos. O que la tecnología referente al hardware, componentes electrónicos no cumpla con las expectativas conceptuales del modelo de datos orientado a objetos.

GLOSARIO

Algoritmo

Un procedimiento del cual se garantiza que concluirá en un tiempo finito

Abstracción

“Acto de separar en el pensamiento”. es la representación de las características esenciales de algo sin incluir detalles de fondo o no esenciales.

Campo

Representa la subdivisión de los datos de un registro

Instancia

Un objeto o ejemplo individual particular de una clase.

Hardware

Equipos físicos como por ejemplo los dispositivos electrónicos magnéticos y mecánicos. Compárese con Software.

Programación Estructurada

Red

Interconexión de sistemas de cómputo o dispositivos periféricos en localidades dispersas que intercambian datos cuando es necesario para llevar a cabo las funciones de la red.

Registro

Representa un elemento determinado de datos

Software

Conjunto de programas, documentos, procedimientos y rutinas
Asociados con la operación de un sistema de cómputo. Compárese con Hardware

Sistema

- 1) Agrupación de métodos y procesamientos integrados para formar un ente organizado.
- 2) Agrupación organizada de personas, métodos máquinas y materiales reunidos para lograr un conjunto de objetivos específicos

Sistemas Operativo

Conjunto organizado de programas que controla las operaciones generales de una computadora

Tabla

Es un medio de almacenar datos que organizan la información dentro de una base de datos. Las tablas tienen una estructura predefinida y contienen datos que se ajustan a esa estructura

La tabla organiza la información en filas y columnas. Dentro de una tabla, una fila se denomina registro, mientras que las filas se denominan campos.

FUENTES DE CONSULTA

JOYANES Aguilar Luis. Programación Orientada a Objetos, Ed Mc Graw Hill 2da.edición Estados Unidos ,1998.

GRAHAM Ian. Métodos Orientados a Objetos, Ed.Addison-Wesley/Diaz de Santos.Estados Unidos. 1996.

DE MIGUEL Adoración, Piattini Mario Fundamentos y modelos de Bases de Datos, Ed. Computec – Ra-Ma.España, 1998

BERTINO Elisa, Martino Lorenzo Sistemas de bases de datos orientados a objetos.Conceptos y arquitectura.Ed. Addison-Wesley/Diaz de Santos Estados Unidos, 1995.

PRESSMAN S. Roger. Ingeniería de Software.Un enfoque Práctico, Ed Mc Graw Hill 4a.edición

PETER M.D Gray, Object Oriented Database.Ed Prentice Hall

BURCH Jhon G , Gary Grudnitski, Diseño de sistemas de información, Ed Grupo Noriega Editores

SILBERTSCHATZ Abraham,Korth Henry F.Fundamentos de Bases de Datos.Ed Mc Graw Hill.España, 1998.

HANSEN Gary W Diseño y Administración de Bases de Datos.Ed. Mc Graw Hill Internacional 1996

LOPEZ Gaona Amparo, El modelo de Objetos de ODMG-93.Soluciones Avanzadas

LOPEZ Gaona Amparo, Lenquajes de ODMG-93.Soluciones Avanzadas

MONTIEL Adriana,Manual de Producción del Módulo de Traspasos Central.Ed. PROCESAR S A de C V. 1997

SANDERS Donald H Informática Presente y Futuro.Ed MC Graw Hill

BOOCH Grady,Análisis y Diseño Orientado a Objetos con Aplicaciones.Ed. Addison-Wesley/Diaz de Santos.Estados Unidos, 1996

McMANUS Jeffrey P.Bases de Datos con Visual Basic.Ed Prentice –Hall.Estados Unidos,1999.