



Diseño de Sistemas basado en TICS

Base de Datos

Bases de datos relacionales y RDBMS
Introducción al Modelo Entidad Relación
Introducción al lenguaje SQL

Aldo Di Biase Friedmann



Diseño de Sistemas basado en TICS

Modelo físico

Atributos y dominios
Relaciones M a M
Tablas y vistas
Índices

Aldo Di Biase Friedmann

Modelo físico

- Específica en todo el detalle necesario la base de datos de la aplicación
- El modelo relacional permite que el modelo físico sea más o menos independiente de la herramienta RDBMS utilizada
 - Diferentes RDBMS tienen diferencias que hacen que el diseño físico no sea 100% independiente de la herramienta
- En la práctica, las mismas herramientas que permiten trabajar con el modelo lógico permiten su detallamiento en el físico

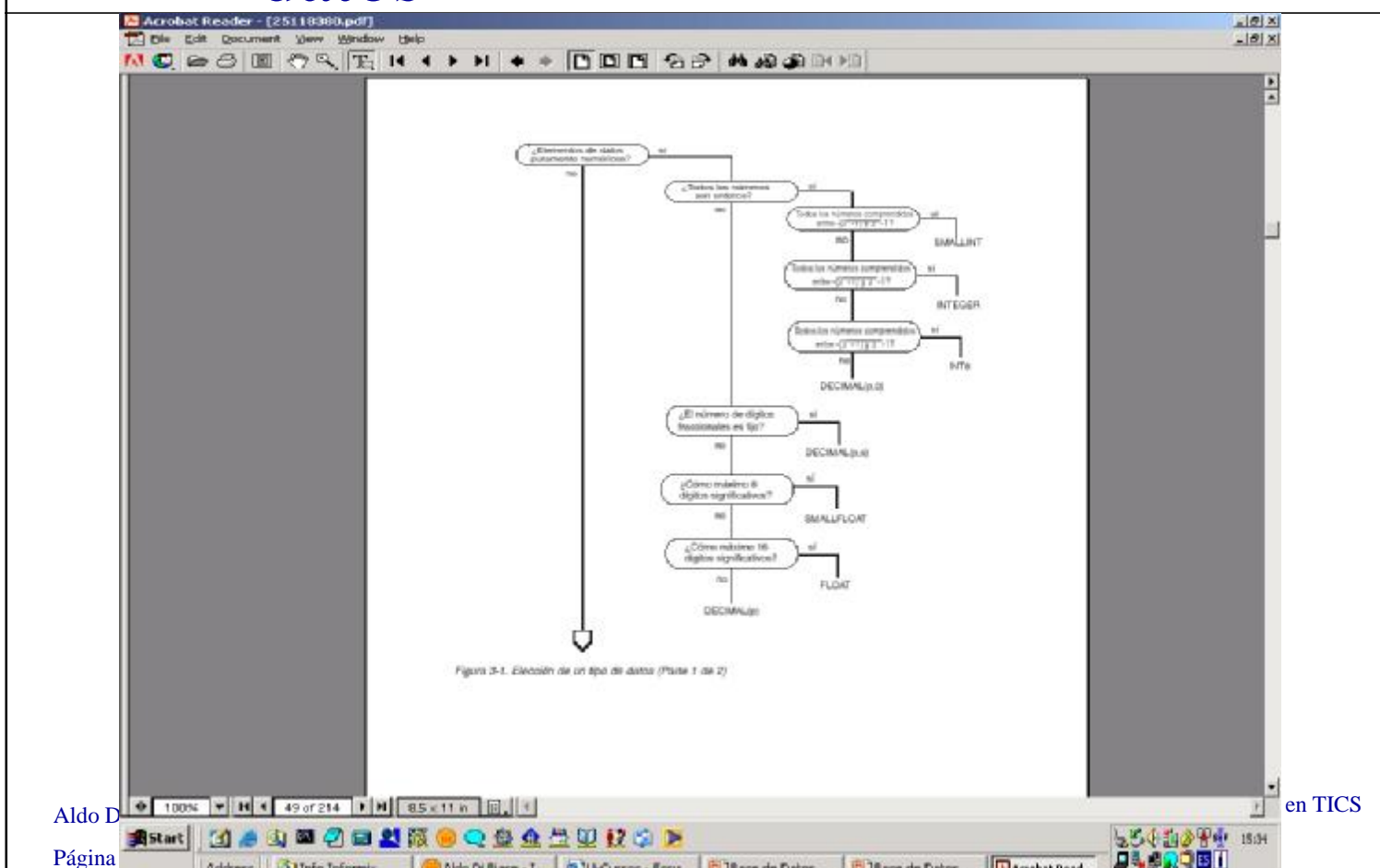
Dominios

- Corresponde a las características de cada uno de los atributos
 - Tipo de dato
 - Largo
 - Total
 - Decimales
 - Restricciones
- Un mismo dominio puede estar asociado a diferentes atributos
 - Ej: Todos los montos pueden tener el mismo dominio

Selección del tipo de datos

- Cada RDBMS tiene su propio conjunto de datos
- Los básicos son
 - String: Fijos y variables
 - Numéricos: Enteros y punto flotante; diferentes tamaños
 - Fecha: Fecha, hora y combinaciones
 - Byte / Text: orientados a grandes volúmenes
- Algunos RDBMS permiten la creación de nuevos tipos
 - Especialización de los anteriores (ej: RUT)
 - Conjunto de campos que se utilizan en conjunto:
 - Rut y dígito
 - Nombres y apellidos
 - Direcciones

Ejemplo para seleccionar el tipo de datos



Otras propiedades de los atributos

- Si son obligatorios o acepta valores nulos
 - Un valor nulo es desconocido (o inexistente); no es lo mismo que 0 ó blanco
- Si puede tener o no un valor repetido
- Debe especificarse en todos sus componentes
 - Nombre: Nombres, apellido paterno, apellido materno
 - Dirección: Calle, N°, Depto, Comuna, Ciudad, Región; Código Postal

Ejemplos de atributos

- Para complementar lo anterior, es una buena práctica colocar ejemplos de cada atributo para un grupo representativo de instancias de la entidad
- En estos ejemplos se deben considerar los valores extremos

Atributos y restricciones

- Un atributo puede tener restricciones intrínsecas
 - Por ejemplo, un porcentaje debe estar entre 0 y 100%
 - En algunos casos estas restricciones pueden ser definidas directamente en el RDBMS; en otros no
- Además, existen restricciones que involucran un grupo de atributos
 - Por ejemplo el total a pagar debe ser mayor que el monto bruto; o la fecha de vencimiento debe ser mayor a la de fabricación
 - En estos casos se deben implementar “triggers”, que son programas (escritos en un lenguaje propio del RDBMS) que se ejecutan al insertar, modificar o borrar un registro
- Los triggers permiten la incorporación de lógica bastante compleja en el propio RDBMS

Atributos y restricciones

- Recomendación:
 - En la medida de lo posible las restricciones se imponen en el RDBMS y no el programa que graba los datos
 - De esta forma, la lógica de las restricciones está en un solo punto (más fácil de administrar y mantener)
 - Además, sea cual fuere la forma de grabar el dato, las restricciones se aplicarán

Tablas y entidades

- Al llevar a un modelo físico el modelo lógico relacional, se tiene que cada entidad corresponde a una tabla
- Cada columna de la tabla corresponde a un atributo

Ejemplo: Tabla de profesores

Apellido	Nombre	Departa- mento	email	Fono
Di Biase	Aldo	DIE	<u>Aldo.di.biase@cl.ibm.com</u>	2006742
Beltran	Nicolás	DIE		6784193

Especificación de entidades (tablas)

- Cada herramienta de diseño tiene su propio formulario; pero en general tiene:
 - Nombre de la entidad
 - Nombre de la tabla
 - Descripción
 - Lista de atributos
 - Nombre
 - Descripción
 - Dominio
 - Restricciones
 - Obligatoriedad
 - Duplicidad

Información multimedia

- Cada vez es más frecuente que este tipo de información se requiera almacenar
- Ejemplos:
 - Manuales
 - Planos
 - Información georeferenciada
 - Páginas Web

Información multimedia

- En general el modelo relacional no está concebido para este tipo de información
- Sin embargo, algunos RDBMS han incluido extensiones para incorporarlos

Información multimedia

- Lo más frecuente es que en la base de datos relacional se almacene:
 - Una referencia al archivo que contiene el documento multimedia
 - Un conjunto de descriptores que ayudan a identificar su contenido
- Se debe tener un directorio (y sus subdirectorios) con esta información
 - La gestión de estos archivos es parte del trabajo de diseño de la base de datos

Llaves foráneas

- Las llaves foráneas (referencias a otras entidades) se colocan como atributos
- El valor del atributo es la llave primaria de la entidad
 - Puede corresponder a más de un atributo
- El RDBMS incorpora estas restricciones, por lo que si se indica una columna como LF, debe existir la LP en la otra entidad (si no se devuelve un error al tratar de grabar)

Llaves foráneas – Relaciones 1 a 1

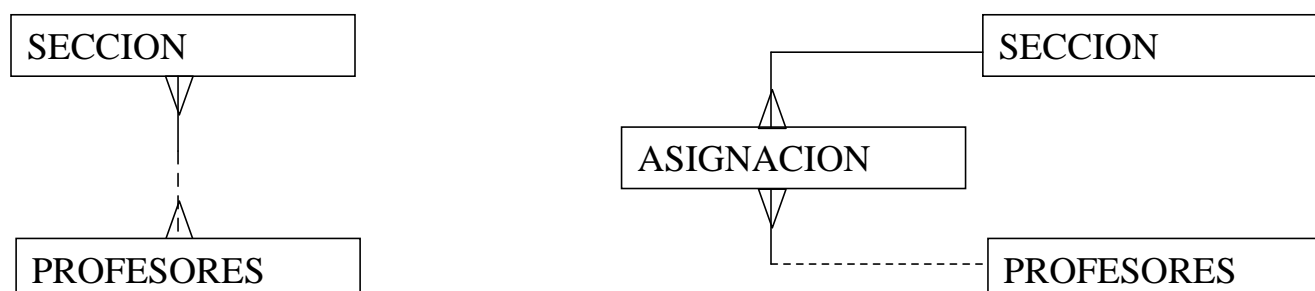
- En general se tiene que una de las dos entidades “depende” de la otra
- En esta entidad dependiente se coloca la LF
- Ejemplo, la esposa de un empleado existe sólo si hay un empleado
 - Luego, la LF se colocar en la entidad Esposa

Llaves foráneas – Relaciones 1 a M

- La LF se coloca en la entidad del lado M
- Ejemplos:
 - Entidades de código
 - Curso y sección

Relaciones M a M

- Las relaciones M a M no se pueden colocar en ninguna de las dos tablas
- Para resolverlo se coloca una tabla intermedia, con relaciones 1 a M con las entidades originales
 - Incluso en muchos casos se coloca en diseño lógico como una entidad



Relaciones M a M

- Al crear la tabla intermedia, se tienen dos relaciones M a 1, que se pueden manejar
- La llave primaria de esta tabla corresponde a dos LF, una a cada entidad
 - Siempre debe ser obligatoria
- Es frecuente que en esta tabla intermedia se coloque información adicional
 - Por ejemplo el tipo de profesor (cátedra, auxiliar, ayudante) de la sección
 - Es común que tengan historia, por lo que se agrega la llave a la LP

Vistas

- Son modificaciones de 1 o más tablas para facilitar su manejo
 - Normalmente se utilizan para tener una recuperación más simple de datos
- No genera un objeto en el RDBMS, sólo una forma de acceder a los datos
- Corresponden a estructuras similares a tablas (de hecho el lenguaje SQL las trata como tablas) formadas con datos de una o más tablas

Usos de vistas

- Decodificación
 - Las tablas contienen el código (LF) a otras entidades, pero en general es bueno visualizar su descripción
 - Se hace el join entre las tablas
 - Se debe tener cuidado de no crear una vista con muchas tablas y luego acceder a sólo algunas
- Campos derivados
 - Totales
 - Partes de las fechas
- Formato para despliegue directo

Índices

- Son estructuras que permiten hacer búsquedas en forma más eficiente en una tabla
 - En algunos casos se usan para ordenar los registros
- Esto porque:
 - Los datos se graban sin un orden específico
 - Se puede querer buscar por diferentes criterios (Rut, nombre)
- Es una de las herramientas usadas para el “tunning” de la base de datos

Selección de índices

- Las LP necesariamente son índices
- Normalmente las LF también
- Cualquier criterio de búsqueda sobre una tabla es un candidato a índice
- Los índices hacen más eficiente la recuperación, pero no así la grabación

Integridad referencial

- ¿Qué pasa si se elimina un registro que está siendo referido en otra tabla?
- No se puede eliminar
- Se elimina en cascada
- Se “nulifica” en cascada
- Lo mismo sucede si se modifica la LP de la entidad
 - Esto debería evitarse; las LP en general no deberían modificarse

Introducción al lenguaje SQL

Creación de la base de datos

Lectura de la base de datos

Modificación de la base de datos

Aldo Di Biase Friedmann

Creación de la base – Create Table

- CREATE TABLE name
(
 rec_num SERIAL PRIMARY KEY,
 lname CHAR(20),
 fname CHAR(20),
 bdate DATE,
 anniv DATE,
 email VARCHAR(25)
);
- CREATE TABLE hijo
(
 child CHAR(20),
 rec_num INT,
 FOREIGN KEY (rec_num) REFERENCES NAME (rec_num)
);

Aldo Di Biase Friedmann

Página N° 28

Diseño de Sistemas basado en TICS

Universidad de Chile

Consulta a la base de datos

- Seleccionar un grupo de campos y algunos registros de una tabla
- Select campos from tabla where condiciones
- Explicación:
 - Campos corresponde a una lista de campos de la tabla indicada
 - Un asterisco (*) indica todos los campos
 - Pueden ser expresiones de campos:
 - Campo 1 + Campo 2
 - Log (campo)
 - Format (campo, formato)
 - Tabla indica la tabla donde están los datos a recuperar
 - Condiciones corresponde a las restricciones de la consulta:
 - Campo = “ABC”
 - Campo 1 = “ABC” AND Campo 2 <> 123
 - Campo 1 = “ABC” OR Campo 1 = “XYZ”
 - Campo 1 between (valor menor, valor mayor)
 - Campo 1 in (select campo, from tabla2 where nueva condición)

Aldo Di Biase Friedmann

Página N° 29

Diseño de Sistemas basado en TICS

Universidad de Chile

Consulta a la base de datos

- Seleccionar un grupo de campos y algunos registros de una tabla y dejarlos ordenados
- Select campos from tabla where condiciones order by campos
- El Order by indica la lista de campos por los que se deben ordenar los registros seleccionados
 - Es una operación “cara”, por lo que debe ser utilizada sólo cuando es necesaria

Aldo Di Biase Friedmann

Página N° 30

Diseño de Sistemas basado en TICS

Universidad de Chile

Consulta a la base de datos

- En este caso se tiene que se recuperan datos desde 2 ó más tablas simultáneamente
- Select campos from tablas where condiciones and join
 - En caso necesario se identifica el campo anteponiendo la tabla:
 - Tabla.campo
 - Las condiciones del where pueden corresponder a campos de cualquiera de las tablas, o sus combinaciones
 - Una condición que se debe colocar es la del Join:
 - Tabla1.LF = Tabla2.LP
 - En caso de no colocarse puede generar problemas (no es una obligación, ya que puede desearse no hacerlo)
 - Existen diferentes tipos de join si se quiere o no incorporar nulos

Consulta a la base de datos

- Pueden recuperarse datos agrupados directamente desde la base de datos
 - Promedios, mayor, menor, cantidad, etc.
- Select campos from tablas where condiciones group by campos having condiciones
- El Group By indica los campos por los que se agrupan los datos
 - Ejemplo: Select Alumno, Average (Notas) from Evaluaciones group by Alumno
- Se pueden agregar condiciones al conjunto para restringir los resultados:
 - Select Alumno, Average (Notas) from Evaluaciones group by Alumno having Average (Notas) >= 4.0 (los aprobados solamente)

Consulta a la base de datos

- La combinación completa de
 - Select **campos** from **tablas** where **condiciones** group by having **condiciones** order by **campos**
- Permite tener una forma muy simple de realizar consultas complejas
 - La gran potencialidad de SQL
 - Puede ser un arma de doble filo, si la base de datos no tiene un buen tuning

Nuevos de registros

- Se usa la instrucción Insert
- Tiene la opción de grabar todos los campos de una tabla o sólo algunos
- Existe la opción que los datos a grabar se obtengan de un select, con lo que se pueden grabar un conjunto de registros en una sola instrucción
 - Muy útil para estadísticas y totales

Eliminación de registros

- Se usa la instrucción Delete tabla where condición
- La condición puede ser tan compleja como lo que se desee (similar a la del Select)

Modificación de registros

- Se usa la instrucción Update tabla set campo 1 = valor, campo 2 = valor where condición
- El valor a grabar puede ser una expresión compleja, no necesariamente un valor simple
- La condición puede ser tan compleja como lo que se desee (similar a la del Select)



Diseño de Sistemas basado en TICS

Base de Datos

Bases de datos relacionales y RDBMS
Introducción al Modelo Entidad Relación
Introducción al lenguaje SQL

Aldo Di Biase Friedmann