

Temario Fundamentos de Bases de Datos

Ismael Sallami Moreno

`ism350zsallami@correo.ugr.es`

`https://ismael-sallami.github.io/`

`https://elblogdeismael.github.io/`

Universidad de Granada

Índice general

1. Tema 1	5
2. Tema 2: Arquitectura de un SGBD	7
2.1. Lenguajes de una BD	7
3. Tema 3: Modelos de datos	9
3.1. Definición	9
3.2. Modelo Relacional	10

Capítulo 1

Tema 1

Capítulo 2

Tema 2: Arquitectura de un SGBD

2.1. Lenguajes de una BD

Recomendación ANSI/SPARC. Se propone un lenguaje de base de datos. En este se debe de definir, controlar y manipular los datos. Se denomina sub-lenguaje de la base de datos al que esta implementado por el propio SGBD. Tenemos distintas partes:

- DDL: definición de estructura de datos.
- DML: modificación, borrado y consulta de datos, además, se permite consultar los esquemas definidos de la Base de Datos.
- DCL: gestiona los requisitos de acceso a los datos y otro tipo de tareas de administración(creación de usuarios,...).

Este grupo propone que haya cada uno de los anteriores en cada uno de los niveles de la arquitectura. Esto no tiene mucho sentido. Esto se debe a que no tiene sentido tener alguno de los anteriores en determinados niveles. Si meto el DDL en el nivel interno, estoy dependiendo del nivel interno de la máquina.

En la realidad ha surgido la idea de que todo sea un estándar, pero cada fabricant lleva sus propias estrategias. Hay diferentes versiones de SQL y los SGBD han ido proporcionando soporte.

Acto seguido, nos centramos en el desarrollo de aplicaciones. Son de propósito general, ya que usan lenguajes como C++, Java,... Se usan diversas herramientas como es Oracle APEX, Sysbase PowerBuilder,... Con esto se proporciona un procesamiento avanzado de datos y una gestión eficaz de la interfaz del usuario.

Debemos de establecer un mecanismo que nos permite establecer una comunicación entre el lenguaje anfitrión y el de datos. Esto se conoce como *acoplamiento*. Distinguimos dos categorías.

- Débilmente acoplados. Lenguajes de propósito general y en este caso el programador puede distinguir entre sentencias del lenguaje anfitrión y las dispuestas por la propia BD.
- Fuertemente acoplados. Lenguajes y herramientas de propósito específico. Se parte del DSL como elemento central y se le van incorporando características para facilitar el desarrollo de aplicaciones.

Hay varias alternativas para implementar el acoplamiento débil:

- **Usar APIs de acceso a la BD.** Acceder a la BD desde el código fuente del lenguaje anfitrión.
- **DSL inmerso en el código fuente del lenguaje anfitrión.** Se escribe código híbrido.

Pasa lo mismo con el acoplamiento fuerte:

Las propuestas son ya propietarias, o bien la ejecución de Java sobre una máquina virtual que esta en el propio SGBD.

También han aparecido numerosos entornos de desarrollo que son específicos para las aplicaciones de gestión.

Capítulo 3

Tema 3: Modelos de datos

3.1. Definición

Previamente, debemos de llevar a cabo un análisis de los datos y de esta manera obtenemos el esquema conceptual y lógico de la BD. En este tema entre la siguiente fase de *Diseño*, dando como resultado el modelo lógico. Este se trata de uno ya implementado, por eso se dice que es un modelo implementable.

En cuanto al proceso de transformación, distinguimos:

1. Mundo real
2. Datos operativos
3. Esquema conceptual

En esta etapa se introduce lo que conocemos como *tablas*. En la última fase llevamos a cabo la implementación de esa tabla mediante un lenguaje.

Definición: Mecanismo formal para representar y manipular la información con la que voy a trabajar. Debe de constar de datos, operaciones y reglas de integridad.

La necesidad de usar el modelo de datos son:

- Se usan lenguajes de definición de datos.
- Es de muy bajo nivel.
- Se necesitan niveles superior para su abstracción.

El objetivo que se tiene es establecer que representen los datos y que los describan de una forma entendible y manipulable. Según ANSI distinguimos tres niveles:

- Externo
- Conceptual
- Interno

En cuanto a la clasificación:

- Basados en registros.
- Basados en objetos.
- Físico.

Los dos primeros son de nivel externo y conceptual, mientras que el último es de nivel interno.

3.2. Modelo Relacional

La información se organiza en tablas:

1. La estructura de almacenamiento son las tablas.
2. Integridad.
3. Consulta y Manipulación.

Esta tabla es a nivel lógico, en cuanto al nivel físico depende de varias estructuras, como son las listas enlazadas,...

Elementos a conocer:

- Atributo.
- Dominio: rango de valores que puede tomar (enteros, letras,...)
- Relación: Se conoce como cualquier subconjunto del producto cartesiano $D_1 \times D_2 \times \dots D_n$. En este caso a relación se refiere a la tabla. Se denota como $R(A_1..A_n)$. De esta manera conseguimos todos los valores que pueden tomar y de aquí sacamos los datos (solo a nivel teórico).
- Tupla: filas de la tabla.
- Cardinalidad de una relación: número de tuplas.
- Esquema de una relación R: Atributos de la relación junto a su dominio.
- Grado de una relación: Número de atributos. Aunque se supone que es invariable, esto depende ya que se puede modificar.
- Instancia de una relación: datos que tengo en un determinado momento.
- Esquema de la BD: colección de esquemas de relaciones junto con las restricciones de integridad.
- Instancia o estado de una BD: colección de instancias de relaciones que verifiquen las condiciones de integridad.
- BD relaciones: instancia de la BD junto con su esquema.

Bibliografía

- [1] Autor(es), *Título del artículo*, Nombre de la Revista, volumen(número), páginas, año.
- [2] Autor(es), *Título del libro*, Editorial, año.
- [3] Autor(es), *Título del documento*, Nombre de la Conferencia, páginas, año.