

Bases de Datos Orientadas a Objetos

Luis Valencia Cabrera (lvalencia@us.es)

Research Group on Natural Computing

Departamento de Ciencias de la Computación e Inteligencia Artificial
Universidad de Sevilla

28-11-2012, Bases de Datos, 3^o Grado en Estadística



Índice

- 1 **Introducción**
 - Introducción
- 2 **Motivación**
 - Motivación
 - Necesidades de los LPOO y limitaciones de las BDR
- 3 **SGBDOO vs. SGBD de tercera generación**
 - SGBDOO vs. SGBD de tercera generación
 - Camino a recorrer. Manifiestos
- 4 **Bibliografía**
 - Bibliografía

Índice

- 1 **Introducción**
 - Introducción
- 2 **Motivación**
 - Motivación
 - Necesidades de los LPOO y limitaciones de las BDR
- 3 **SGBDOO vs. SGBD de tercera generación**
 - SGBDOO vs. SGBD de tercera generación
 - Camino a recorrer. Manifiestos
- 4 **Bibliografía**
 - Bibliografía

Introducción (I)

La industria del software experimenta un importante crecimiento durante los 80 y los 90. Junto al paradigma de la **programación estructurada**, comienza a cobrar una importancia creciente la **programación orientada a objetos**.

Junto a los **tipos de datos primitivos** (*int*, *char*, *bool*, *float*, etc.) y a estructuras de datos más o menos sencillas (*arrays*, cadenas), aparecen niveles de organización de orden superior, centrando la atención en los **tipos abstractos de datos** (*pilas*, *montículos*, etc.), desembocando en la aparición de conceptos como clases (*Persona*), interfaces (*Visitable*), etc. con el objetivo de representar una realidad cada vez más compleja en los sistemas de información.



Introducción (II)

Los requisitos de los sistemas de información vienen siendo **cada vez más complejos**, manejando una importante cantidad de nociones, conceptos del mundo real, y requiriendo *desarrollos más rápidos* y seguros, lo que generalmente implicará la necesidad de trabajar a más alto nivel.

Además, la **información a almacenar** en los sistemas de información también crece en complejidad, pudiendo necesitarse almacenar **gráficos, vídeo, audio, diagramas, huellas digitales**, así como los tradicionales *números, textos*, etc.



Introducción (III)

Los modelos y sistemas de **bases de datos tradicionales**, fundamentalmente los relacionales, han tenido un **gran éxito** en el desarrollo de las bases de datos requeridas por muchas de las **aplicaciones de bases de datos tradicionales**. Sin embargo, tienen **ciertas carencias** cuando se deben diseñar e implementar **aplicaciones de bases de datos más complejas** (por ejemplo, bases de datos para aplicaciones de ingeniería del diseño y fabricación como los *CAD/CAM/CIM*, experimentos *científicos*, *telecomunicaciones*, *SIG*, *multimedia*, *CASE*, *documentales*, etc).

Surge por tanto la necesidad de **proporcionar alternativas** que puedan adaptarse a **escenarios más complejos**, fruto tanto de la **creciente complejidad en los requisitos de las nuevas aplicaciones** y sistemas software, como del almacenamiento de tipos de información cada vez más complejos. Ambas necesidades llevan a mirar hacia delante para subsanar las carencias de las bases de datos relacionales en ciertos escenarios.



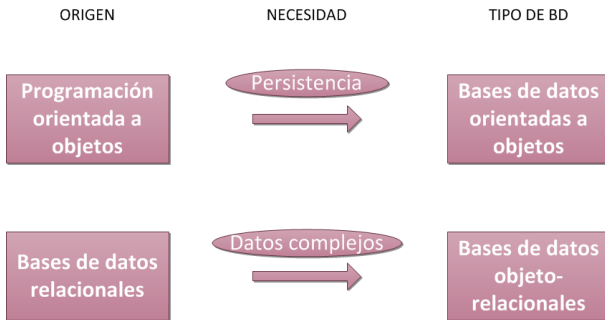
Índice

- 1 Introducción
 - Introducción
- 2 Motivación**
 - Motivación
 - Necesidades de los LPOO y limitaciones de las BDR
- 3 SGBDOO vs. SGBD de tercera generación
 - SGBDOO vs. SGBD de tercera generación
 - Camino a recorrer. Manifiestos
- 4 Bibliografía
 - Bibliografía

Motivación (I)

Motivación. ¿Por qué surgen las bases de datos orientadas a objetos?

- Por necesidades de los lenguajes de programación OO.
- Por limitaciones de las BD relacionales.



Motivación (II)

Problemas

- 1 ¿Qué ocurre si convertimos objetos y relaciones al modelo relacional?
 - Un objeto se descompondrá en gran número de tablas \Rightarrow errores.
 - Se necesitará un gran número de joins para su recuperación \Rightarrow empeoramiento del rendimiento.

Solución: SGBD relacionales de objetos.

- 2 Los modelos de datos y las estructuras de datos de los LP está desacoplados:
Solución: SGBDOO puras, de modo que ambos sigan el mismo paradigma.
Esta aproximación engloba la gestión tanto de los datos como de su comportamiento.
Incluye los lenguajes ODL y OQL.



Necesidades de los LPOO

Las BD proporcionan a la OO la **Persistencia**

- **Eficiente** almacenamiento y gestión de datos en memoria secundaria.
- **Independencia** de los datos respecto de los programas.
- Lenguaje de **consulta** eficiente y de alto nivel (independiente de la estructura física).
- Gestión de **transacciones** que permita: acceso **concurrente**, **seguridad** y **recuperación** ante fallos.
- Control de **integridad** (*restricciones, aseveraciones y disparadores*).

Limitaciones de las bases de datos relacionales

Limitaciones de las BDR

- Presentan **estructuras simples** (ej: imposición de 1FN).
- **Poca** riqueza **semántica**.
- **No** soportan **tipos** definidos por el **usuario** (sólo dominios).
- **No** soportan **recursividad**.
- Falta de **procedimientos**/disparadores.
- **No** admite **herencia**.

Por todo lo anterior, las bases de datos relacionales no se consideran apropiadas para aplicaciones que manejen estructuras de datos complejas.

Necesidades de las nuevas aplicaciones

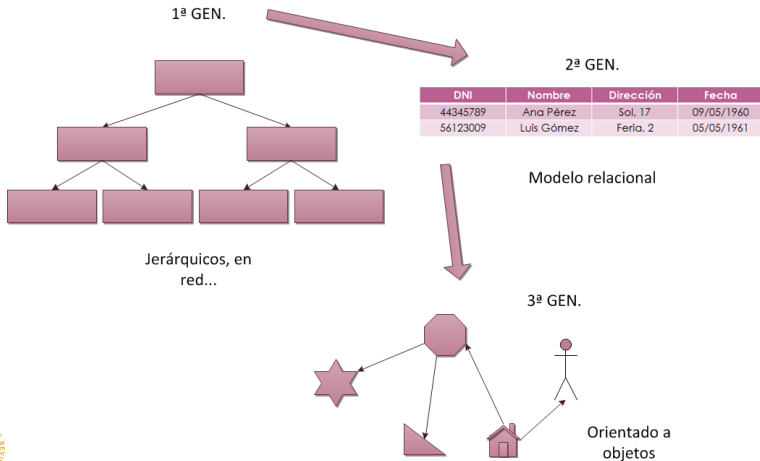
Algunas nuevas necesidades

- Soporte de objetos **complejos** y datos **multimedia**.
- **Identificadores** únicos.
- Soporte de **referencias** e **interrelaciones**.
- Manipulación **navegacional** y de **conjunto** de registros.
- **Jerarquías** de objetos y **herencia**.
- Integración de los datos con sus **procedimientos** asociados.
- Modelos extensibles mediante **tipos de datos definidos por el usuario**.
- Gestión de **versiones**.
- Facilidades de **evolución**.
- **Transacciones** de larga duración.
- **Interconexión** e **interoperabilidad**.

Índice

- 1 Introducción
 - Introducción
- 2 Motivación
 - Motivación
 - Necesidades de los LPOO y limitaciones de las BDR
- 3 **SGBDOO vs. SGBD de tercera generación**
 - SGBDOO vs. SGBD de tercera generación
 - Camino a recorrer. Manifiestos
- 4 Bibliografía
 - Bibliografía

Generaciones de bases de datos



SGBDOO

Visión SGBD



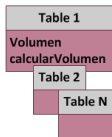
Visión Programa

```

public static double CalcularVolumen(Double radio)
{
    double volumen=0;
    volumen=Math.pow(radio,3);
    return volumen;
}

public static void main(String[] args)
{
    double volumen=0,radio=0;
    if(args.length==0)
    {
        System.out.println("Tiene que recibir como valor el radio de la
esfera");
    }else{
        System.out.println("El Volumen de una esfera con radio '"+args[0]+"' es:
"+CalcularVolumen(Double.parseDouble(args[0]));
    }
}

```



Visión SGBDOO



SGBDOR

Visión SGBD Relacional

APLICACIONES	Código de aplicación
SGBD	Estructuras de datos simples

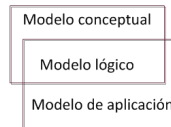
Visión SGBD 3ª Gen.

APLICACIONES	Código de aplicación
SGBD	Estructuras de datos complejas
	Operaciones sobre datos

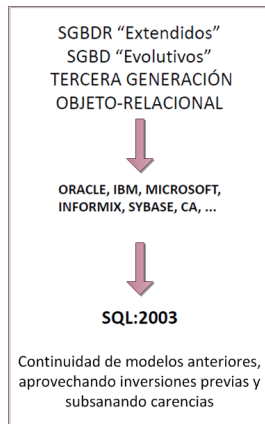
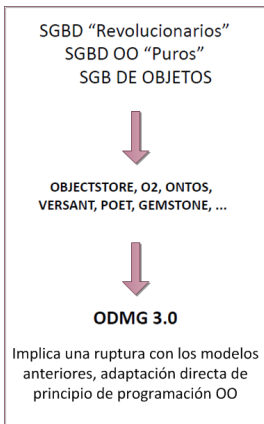
Modelo conceptual



Modelo lógico de datos + Modelo de aplicación



Enfoques de implementación de SGBD de Objetos



Manifiesto de los SGBDOO (I)

Atkinson, Bancilhon, DeWitt. Dittrich, Maier, Adonik(1989)

Tipos de características

- Obligatorias: **imprescindible** satisfacerlas para ser calificadas como OO.
- Opcionales: **pueden añadirse** para **mejorar** el sistema.
- Abiertas: posibilidades adicionales **aceptables**, a aplicar a juicio del diseñador.

Manifiesto de los SGBDOO (II). Características obligatorias

Al ser un SGBD

- Persistencia.
- Gestión del almacenamiento secundario.
- Concurrencia.
- Recuperación ante fallos.
- Lenguajes ad-hoc para manipulación.

Al ser OO

- **Objetos complejos.**
- **Identidad del objeto.**
- Encapsulamiento.
- **Tipos o clases.**
- Herencia.
- Polimorfismo, sobrecarga y vinculación dinámica.
- **Extensibilidad.**
- Completitud de cálculos (lenguaje de propósito general).

Manifiesto de los SGBDOO (III). Características opcionales y abiertas

Características opcionales

- Herencia múltiple.
- Verificación e inferencia del tipo.
- Distribución.
- Transacciones de diseño.

Opciones abiertas

- Paradigma de programación.
- Sistema de representación (tipos atómicos y constructores).
- Sistema de tipos.
- Uniformidad (¿todo objetos?).

Manifiesto de los SGBD de 3^a Generación (I)

Stonebraker, Lindsay, Gray, Carey, Brodie, Bernstein, Beech (1990).

Principio 1

*Además de los servicios tradicionales de gestión de datos, los SGBD-3G proporcionarán **gestión de objetos y reglas más ricas.***

- 1 Un SGBD-3G debe tener un **sistema de tipos rico.**
- 2 La **herencia** es una buena idea.
- 3 Las **funciones** (procedimientos y métodos) son una buena idea.
- 4 Los **IDO**s para los registros deberían asignarse por el SGBD **sólo si no se dispone de una clave primaria.**
- 5 Las reglas (disparadores, restricciones) se convertirán en una característica primordial de los sistemas futuros.



Manifiesto de los SGBD de 3^a Generación (II)

Principio 2

Los SGBD-3G deben subsumir los SGBD-2G.

- 1 **Lenguaje de acceso declarativo** (no procedimental) y de **alto nivel**.
- 2 Dos formas de especificar **colecciones**: enumeración de miembros y lenguajes de consulta para especificar la condición de pertenencia.
- 3 Las **funciones** (procedimientos y métodos) son una buena idea.
- 4 **Vistas** actualizables.
- 5 Los indicadores de **rendimiento no** deben aparecer en los **modelo de datos**, ya que no tiene prácticamente nada que ver con los modelos de datos.

Manifiesto de los SGBD de 3^a Generación (III)

Principio 3

Los SGBD-3G deben ser abiertos a otros subsistemas.

- Los SGBD-3G deben ser accesibles desde múltiples lenguajes de alto nivel.
- Persistencia de variables.
- El **SQL** es una forma intergaláctica de expresión de datos.
- Las **consultas** y las **respuestas** resultantes deben ser el **nivel más bajo de comunicación** entre un cliente y un servidor 3.2 M.

Productos y estándares. SGBDOO puros

Estándares

ODMG-93, Cattell(1994), Cattell(1995), ODMG V.2.0 Cattell(1997), ODMG V.3.0 Cattell(2000).

Productos

- ObjectStore de Object Design. Persistencia de objetos en C++, Java.
- O2 de O2, Leeluse et al. (1988). Lenguajes: C++, lenguajes de consulta (O2SQL) y programación (O2C) propios. Java.
- Gemstone de Servi Logic, Meier y Stone (1987) Persistencia de objetos en SamalltalkSoporta también C++ y Java.
- POET de Poet Corporation Persistencia de objetos C++, Java.
- db4o, Matisse...



Productos y estándares. SGBD Objeto-Relacional

Estándares

SQL: 1999, Melton (1999). SQL: 2003, Melton (2003).

Productos

- POSTGRES (Miró/Illustra), Stonebraker et al. (1992) Combina capacidades de BD OO y activas con BD relacionales.
- ORACLE V8, de Oracle (1997) Extiende el modelo relacional del SQL92 con capacidades de objetos y actividad.
- Universal Server de Informix, ...

Convergencia

Concepto

Tratar de evitar a las aplicaciones cliente el dilema entre los distintos tipos de SGBD, puesto que las bases de datos terminarán capturando la esencia y complejidad de la OO, manteniendo las ventajas del modelo relacional.

Poder disponer de un programa tradicional o de un programa orientado a objetos, y que en cualquier caso se puedan comunicar indistintamente con bases de datos relacionales u orientadas a objetos.

Convergencia en cuanto a estándares

OBJECT MERGER GROUP. Grupo formado por integrantes del ODMG y del SQL3 cuyo objetivo es lograr la integración de los lenguajes de consulta de ambos estándares, a fin de conseguir el entendimiento entre BD3G y BDOO.

Convergencia en cuanto a productos

UniSQL. Permite la coexistencia entre BD relacionales y jerárquicas, BD relacionales y BD orientadas al objeto.

Índice

- 1 **Introducción**
 - Introducción
- 2 **Motivación**
 - Motivación
 - Necesidades de los LPOO y limitaciones de las BDR
- 3 **SGBDOO vs. SGBD de tercera generación**
 - SGBDOO vs. SGBD de tercera generación
 - Camino a recorrer. Manifiestos
- 4 **Bibliografía**
 - Bibliografía

Bibliografía

- **Bases de Datos Orientadas a Objetos**, Departamento de Lenguajes y Sistemas Informáticos II, Escuela Superior de Ingeniería Informática, Universidad Rey Juan Carlos (2011-2012).
[http://www.kybele.etsii.urjc.es/docencia/BD/2012-2013/Material/\[BD-2011-12\]T1.BDOO.pdf](http://www.kybele.etsii.urjc.es/docencia/BD/2012-2013/Material/[BD-2011-12]T1.BDOO.pdf)
- **Bases de Datos Avanzadas, Modelo de Objetos, Objetos Puro**, Francisco Ruiz, Marta Zorrilla. Facultas de Ciencias, Universidad de Cantabria.
<http://personales.unican.es/zorrillm/BDAvanzadas/Teoria/bda-t3-trans-orientadasObjeto.pdf>

