Introducción

Acceso a Datos

Alejandro Roig Aguilar
alejandro.roig@iesalvarofalomir.org

IES Álvaro Falomir
Curso 2023-2024

Programas y datos

Aplicación informática = programa + datos

Un programa realiza las operaciones deseadas con los datos necesarios.

Estos datos pueden ser obtenidos mediante diversos métodos:

- leídos mediante teclado,
- escaneados,
- leídos de algún soporte de almacenamiento,
- etc.

Soportes de almacenamiento de datos

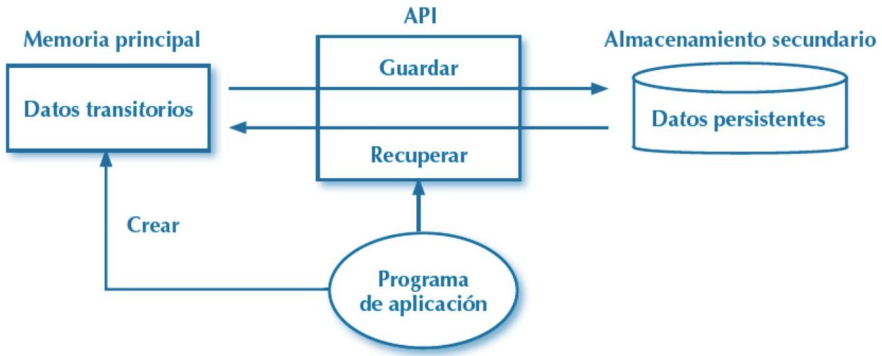
Dos tipos de medios de almacenamiento de datos:

- Almacenamiento primario o memoria principal: almacena los datos con los que está trabajando el programa.
  - Sus contenidos se borran cuando finaliza su ejecución.
  - Capacidad baja y tiempo de acceso muy corto
- Almacenamiento secundario: almacena los datos de manera permanente. Ej: discos duros, memorias flash...
  - No se borran cuando finaliza su ejecución.
  - Capacidad alta y tiempo de acceso largo

Persistencia de datos

Los programas solo pueden consultar (y crear) directamente datos almacenados en almacenamiento primario, llamados datos transitorios.

Generalmente, interesa que el programa guarde datos de manera que si termina su ejecución, los datos no se pierdan y puedan ser recuperados posteriormente, es decir, sean datos persistentes.



Sistemas de persistencia de datos

- Ficheros
- Bases de datos, que pueden ser:
  - Relacionales
  - Orientadas a objetos
  - NoSQL
    - Documentos
    - Clave-valor
    - Columnas
    - Grafos

Ficheros

Es una secuencia de bytes almacenados en un dispositivo.

Se identifica por:

- Nombre: cómo se llama el fichero
- Extensión: qué tipo de fichero es
- Ruta: dónde se encuentra ubicado el fichero

Un fichero debe tener un nombre único en su ruta, pero pueden existir dos ficheros con el mismo nombre en rutas diferentes.

Ficheros

Ejemplo de fichero:

Francisco Pérez Gómez C/ Mayor 27 Borriol Castellón
Juan Bueno Hernández C/ Colón 10 Valencia Valencia
Arturo Marín Carrasco Plz Ayuntamiento 6 Alaquas Valencia
Mark Jones Camino la Ralla s/n Alcázar de San Juan Ciudad Real

El programador que usa este fichero construye el programa conociendo detalladamente las posiciones de los datos.

Ficheros

Otros inconvenientes:

- Rendimiento pobre con volumen de datos grande o si se realizan operaciones de borrado o modificación con frecuencia.
- Concurrencia de aplicaciones que requiere la necesidad de mecanismos de control de acceso.
- Complejidad para evitar la redundancia e inconsistencia en los datos.
- Complejidad para definir y preservar restricciones de integridad.

Ficheros

Actualmente se utilizan en aplicaciones para guardar información simple como un fichero de configuración o un fichero log.

Para almacenar información que siga un patrón o una estructura bien definida, el uso de ficheros puede tener sentido.

Ejemplos:

- CSV
- XML
- JSON

Bases de datos

Un sistema de bases de datos es:

- Un sistema de información orientado hacia los datos, que pretende recuperar y almacenar la información de manera eficiente y cómoda.
- Surge en un intento de resolver las dificultades del procesamiento tradicional de datos, teniendo en cuenta que los datos suelen ser independientes de las aplicaciones.

Bases de datos relacionales

Las bases de datos relacionales organizan los datos en tablas y permiten especificar las relaciones entre dichas tablas.

EMPLEADOS

num_emp	nom_emp
7643	GIL SAAVEDRA, FERNANDO
8257	MARTÍN FERRIS, MARÍA
9087	BORGES RULFO, JUAN

PROYECTOS

id_proy	nom_proy
2	RUEDAS CUADRADAS
5	PAPEL ELECTRÓNICO
8	PERMUTADOR CUÁNTICO

ASIG_PROYECTOS

num_emp	id_proy
7643	2
8257	2
8257	5

Bases de datos relacionales

- Son las más extendidas en la actualidad.
- Son muy robustas, al estar basadas en los fundamentos matemáticos del modelo relacional.
- Permiten interoperabilidad entre aplicaciones y tecnologías.
- Para su uso existe un lenguaje estándar y universal: SQL.
- Son muy escalables.
- Soporte para transacciones -> Propiedades ACID
- Mecanismos de copia de seguridad y recuperación ante fallos.

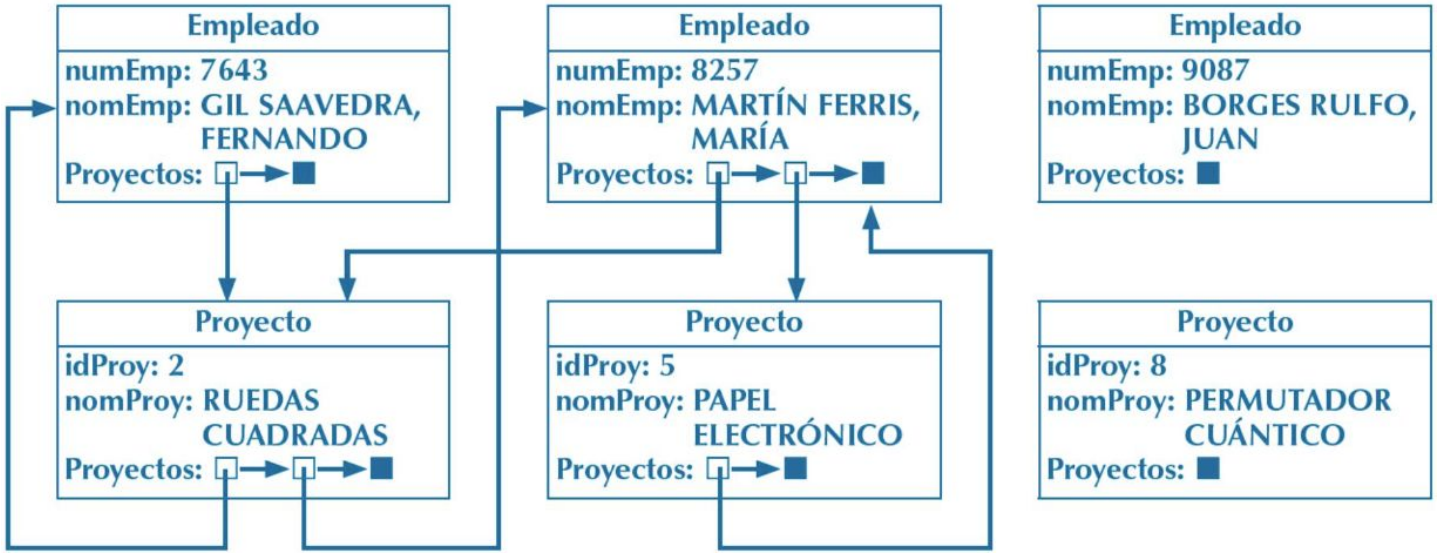
Bases de datos de objetos

El origen de las bases de datos de objetos (BDOO) se debe a:

- La existencia de problemas al representar cierta información y modelar ciertos aspectos del mundo real. Los modelos clásicos permiten representar gran cantidad de datos, pero las operaciones y representaciones que se pueden realizar sobre ellos son bastante simples.
- Pasar del modelo de objetos en que programamos al modelo relacional en que almacenamos la información genera dificultades. En las BDOO, los datos de los programas escritos en lenguaje orientado a objetos se pueden almacenar directamente.

Bases de datos de objetos

Las BDOO almacenan objetos que pueden incluir referencias a otros objetos y a colecciones de objetos relacionados con él. Una colección de objetos tiene estructura de grafo.



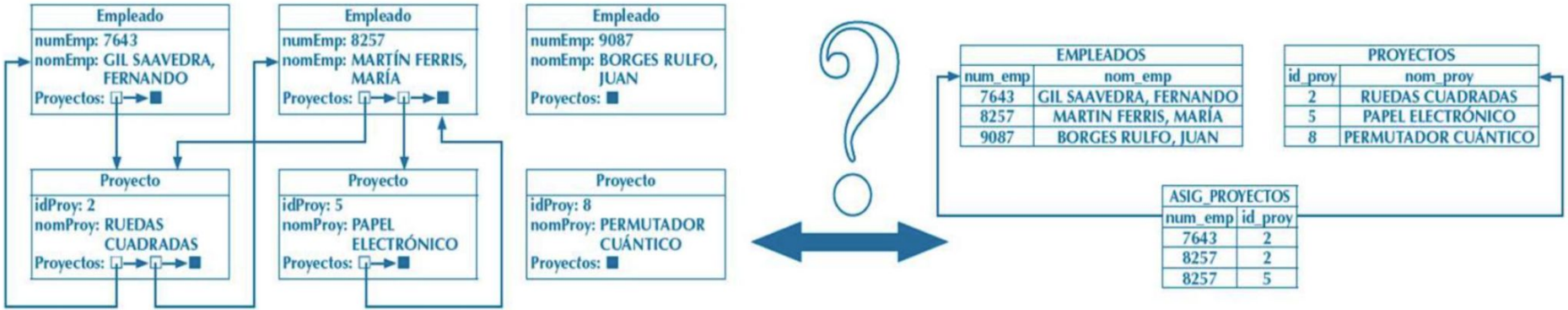
Bases de datos de objetos

La persistencia de objetos en BDOO es una solución natural, sin embargo, las BDOO tienen ciertos inconvenientes:

- La falta de un modelo formal y ampliamente aceptado en el que se basarse, a diferencia de las BD relacionales.
- La falta de estándares ampliamente adoptados, como es el caso del lenguaje SQL en las BD relacionales.

¿ENTONCES?

Desfase objeto-relacional



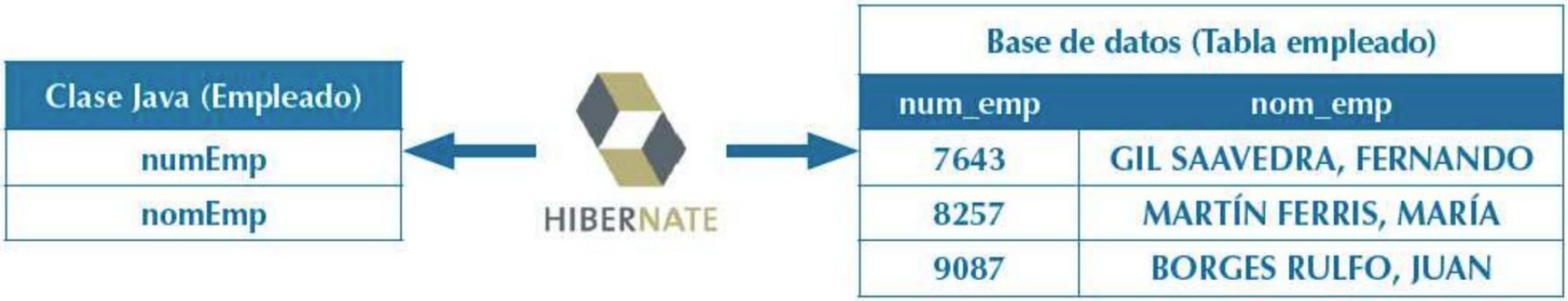
Desfase objeto-relacional

La solución es plantear la persistencia de objetos utilizando bases de datos relacionales.

Esto conlleva un conjunto de problemas a resolver denominado desfase objeto-relacional, ante los que se han planteado dos soluciones:

- BD objeto-relacionales: Son BD relacionales con capacidad para gestionar objetos. Destacan Oracle y PostreSQL.
- Mapeo objeto-relacional (ORM): Es una solución más flexible con la ventaja de proporcionar soporte para múltiples BD. Existen múltiples herramientas, bibliotecas o frameworks para ORM, entre las que destaca Hibernate.

Mapeo objeto-relacional (ORM)



Bases de datos NoSQL

El auge de las BD NoSQL se debe a la necesidad de recopilar, gestionar y analizar gigantescos conjuntos de datos heterogéneos, que crecen exponencialmente → BIG DATA

Este tipo de BD dan respuesta a las necesidades de aplicaciones que ofrecen servicio, a través de la web, a cada vez más usuarios que no solo consultan información, sino que la añaden y la modifican de manera continua.

Bases de datos NoSQL

Las BD NoSQL dan respuesta a la necesidad de ofrecer servicio a cada vez más usuarios que no solo consultan información, sino que la añaden y la modifican de manera continua.

Características de las bases de datos NoSQL:

- El almacenamiento se basa en estructuras flexibles como arrays asociativos (Redis), documentos (MongoDB), etc.
- No se manejan con lenguajes como SQL, sino propios.
- Frente a las transacciones ACID del modelo relacional, prima la disponibilidad, resumido con el acrónimo BASE (disponibilidad básica, estado flexible y consistencia con el tiempo).

¿Qué es un componente?

Hace referencia a cualquier componente software que explote una fuente de información (BBDD, ficheros, etc.)

Es fundamental plantear el diseño y desarrollo teniendo en cuenta su ciclo de vida, securización, testeo y despliegue. Requiere de aplicar patrones de diseño y el uso de arquitecturas en capas.

- Controladores: reciben peticiones de usuarios y devuelven respuestas.
- Servicios: implementan la parte de negocio o infraestructura.
- Repositorios: implementan la interfaz y operaciones de persistencia de la información.

¿Qué es un componente?