

Estructuras de Datos

Curso 2011 / 12

Pablo Castells, Francisco de Borja Rodríguez, Simone Santini

Alejandro Bellogín, Ana González, José Miguel Martínez, Alvaro del Val

Escuela Politécnica Superior

Universidad Autónoma de Madrid

Guión

- ◆ Presentación de la asignatura
 - Motivación y contexto de la materia
 - Aspectos formales y organizativos
- ◆ Introducción a la materia
- ◆ SQL

Datos generales de la asignatura

- ◆ Profesor del grupo 127 – Pablo Castells
 - Despacho B-415
 - Tutorías: por cita a petición del estudiante
- ◆ Profesores de la asignatura
 - Teoría: Pablo Castells, Francisco de Borja Rodríguez, Simone Santini
 - Prácticas: Alvaro del Val, Alejandro Bellogín, Ana González, José Miguel Martínez
- ◆ Horario
 - Martes 17 a 19h
 - Miércoles 17 a 18h
- ◆ Prueba final
 - Lunes 9 de enero, 10h

¿Dé qué trata la asignatura?

Un primer curso en **bases de datos**

¿Qué es una base de datos?

¿Qué es una base de datos?

- ◆ Una fuente de información estructurada almacenada en memoria secundaria
- ◆ Habitualmente masiva en volumen de datos, variedad y complejidad de las estructuras
- ◆ Gestionada mediante tecnologías con un alto nivel de generalidad, desarrolladas y estandarizadas al efecto
- ◆ En la asignatura estudiaremos:
 - Tecnologías de gestión de bases de datos –SQL y nociones prácticas
 - Una introducción a las metodologías de diseño –modelo E/R
 - La base formal sobre la que se asientan las tecnologías de bases de datos –modelo relacional, formas normales, cálculo y álgebra relacional
 - La implementación de tecnologías de gestión de bases de datos –registros, índices, árboles B, hashing, compresión

Un ejemplo

- ◆ Supongamos que vamos a desarrollar una aplicación para escuchar música, con redes sociales
- ◆ Queremos manejar información sobre: canciones, artistas, álbumes, usuarios, grupos, eventos, registro de accesos a canciones...
- ◆ Esta información:
 - Tiene estructura: p.e. un usuario tiene propiedades (nombre, nick, email, etc.) y relaciones (amigos, artistas favoritos, escuchas a canciones, etc.)
 - Tiene que almacenarse de forma persistente
 - Almacenamiento en disco
 - Puede ser masiva: millones de usuarios, millones de canciones, miles de millones de registros de escuchas
 - Inviabile carga completa en RAM
 - Acceso continuo a disco en tiempo de ejecución

- ◆ Para la aplicación necesitamos:
 - Acceso (consulta) eficiente a los datos: mostrar a un usuario su lista de amigos, la información de una canción, etc.
 - Actualización eficiente de datos: añadir amigos, guardar un log de escuchas, etc.
 - Acceso concurrente, robustez, seguridad...
- ◆ Solución: guardar todos los datos en fichero(s) en disco y programar la funcionalidad de acceso
 - No perder la estructura de los datos
 - El acceso a memoria secundaria es muy costoso
 - El problema requiere soluciones elaboradas no triviales (las estudiaremos!)
- ◆ En rigor esto ya se podría considerar una base de datos!

- ◆ La dificultad y complejidad de un desarrollo desde cero son considerables, y por otro lado...
 - ◆ Una buena parte del problema a resolver se repite en muchos dominios: gestión de personal, gestión de inventarios, gestión universitaria, bibliotecas, reserva de viajes, banca, finanzas, contabilidad, competiciones deportivas, correo electrónico, buscadores web...
 - Diseño de las estructuras: tabular
 - Almacenamiento físico
 - Consulta y actualización
 - Integridad, robustez, concurrencia, seguridad...
- } Lo estudiaremos (implementación)
- ◆ Tecnología de bases de datos
- | | | | | |
|--------------------|---|--|---|------|
| Herramientas | { | <ul style="list-style-type: none"> – Lenguaje estándar de creación, consultas y actualización: SQL – Motores de ejecución eficiente de las sentencias SQL – Interfaz de usuario – Interfaz de programación (ODBC, JDBC, PHP) | } | SGBD |
| Método
y teoría | { | <ul style="list-style-type: none"> – Metodologías de diseño: modelo Entidad / Relación – Paradigmas: modelo relacional, formas normales, cálculo y álgebra | } | |

Arquitectura ad hoc

Usuario final

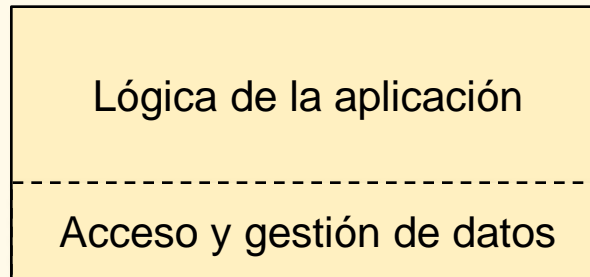
Interfaz de usuario



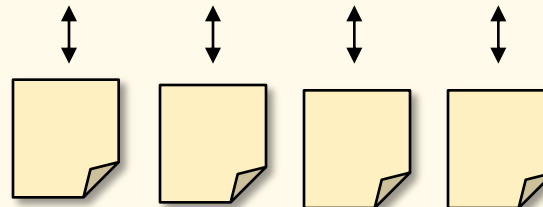
Software aplicación



Programador aplicación



Archivos de datos
Memoria externa



Usuario final

Interfaz de usuario



Programador aplicación

Arquitectura BD

Software aplicación

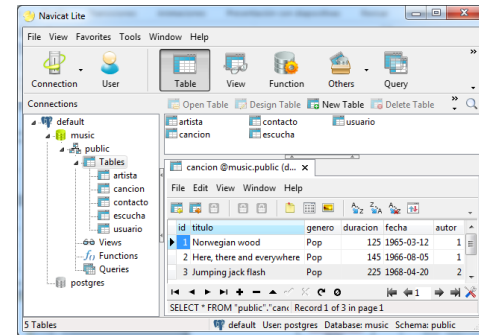


Administrador

Lógica de la aplicación

Entorno / herramientas de administración

Sentencias SQL



Navicat SQLYog etc.

ODBC, JDBC, PHP...

API BD

Almacenamiento
Consultas
Actualización

SGBD

Bases de datos



- ◆ La dificultad y complejidad de un desarrollo desde cero son considerables, y por otro lado...
 - ◆ Una buena parte del problema a resolver se repite en muchos dominios: gestión de personal, gestión de inventarios, gestión universitaria, bibliotecas, reserva de viajes, banca, finanzas, contabilidad, competiciones deportivas, correo electrónico, buscadores web...
 - Diseño de las estructuras: tabular
 - Almacenamiento físico
 - Consulta y actualización
 - Integridad, robustez, concurrencia, seguridad...
- } Lo estudiaremos (**implementación**)
- ◆ Tecnología de bases de datos
- | | | | | |
|--------------------|---|--|---|------|
| Herramientas | { | <ul style="list-style-type: none"> – Lenguaje estándar de creación, consultas y actualización: SQL – Motores de ejecución eficiente de las sentencias SQL – Interfaz de usuario – Interfaz de programación (ODBC, JDBC, PHP) | } | SGBD |
| Método
y teoría | { | <ul style="list-style-type: none"> – Metodologías de diseño: modelo Entidad / Relación – Paradigmas: modelo relacional, formas normales, cálculo y álgebra | } | |

Temario

- ◆ Introducción y fundamentos
- ◆ Introducción a SQL
- ◆ Modelo Entidad / Relación
- ◆ Modelo relacional
- ◆ Diseño relacional: formas normales
- ◆ Consultas: cálculo y álgebra relacional
- ◆ Implementación de bases de datos
 - Estructura física: campos y registros
 - Indexación: índices simples, árboles B, hashing
 - Compresión

“Outcomes”

Las BDs son una área fundamental en la formación de un informático y en la industria del desarrollo de software

Nivel práctico

- ◆ Uso / administración básica de BDs
- ◆ Análisis, diseño y creación de BDs
- ◆ Manipulación de BDs, consultas
- ◆ Manejo de SQL

Nivel conceptual

- ◆ Comprensión de los principios y formalismos sobre los que se asientan las tecnologías de BDs

Nivel técnico

- ◆ Conocimiento de las técnicas de implementación interna de un SGBD

Bibliografía

- ☞ Fundamentos de sistemas de bases de datos. Ramez Elmasri, Shamkant Navathe. Pearson Addison Wesley, 2007. INF/681.31.65/ELM.
- ◆ Database Management Systems. Raghu Ramakrishnan, Johannes Gehrke. McGraw-Hill, 2003. INF/C6160/RAM.
- ◆ Database Systems: The Complete Book. Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom. Prentice Hall, 2008.
- ◆ Fundamentos de diseño de bases de datos. Abraham Silberschatz. McGraw-Hill, 2007. INF/681.31.65/SIL.
- ☞ Estructuras de archivos: un conjunto de herramientas conceptuales. Michael J. Folk, Bill Zoellick. Addison-Wesley, 1992. INF/681.3.01/FOL.

Relación con el plan de estudios

- ◆ Programación I y II, Análisis de algoritmos
 - Programación y desarrollo de software → técnicas específicas para almacenamiento y acceso a datos estructurados masivos en disco
 - Algoritmia en RAM → revisión para datos en disco
- ◆ Análisis y Diseño de Software
 - Modelado de datos: UML
- ◆ Sistemas Informáticos I
 - Optimización de consultas, interfaces de programación
 - Bases de datos distribuidas
 - Transacciones
- ◆ Ingeniería del Software
 - Las BDs son una pieza básica en el análisis y diseño de aplicaciones

Evaluación

60% Teoría ≥ 5	40% Evaluación continua	25% Pruebas ejercicios
		75% Examen intermedio
	60% Examen final (9 enero)	
40% Prácticas ≥ 5 (cada práctica ≥ 3)		

Grupos de prácticas

- ◆ Apuntarse en Moodle
 - <https://moodle.uam.es/course/view.php?id=4208>
 - Password: epsuam
 - Antes del miércoles 14 sep
- ◆ Grupos de prácticas
 - Preferencia + equilibrado aleatorio
 - Grupo teoría → grupos de prácticas correspondientes
 - Horarios complementarios con Análisis de Algoritmos