Databases

0

Introduction

Data bases 2

Instructors: Sara Comai, Piero Fraternali
sara.comai@polimi.it
piero.fraternali@polimi.it

"Data bases 2"
Academic Year 2020/2021

- Teachers:
  - Prof. **Sara Comai** sara.comai@polimi.it 02-2399-3611
  - Prof. **Piero Fraternali** piero.fraternali@polimi.it 02-2399-3640

  @Dipartimento di Elettronica Informazione e Bioingegneria, 1st floor

- Exercise sessions:
  Prof. Daniele Braga, Piero Fraternali, Sara Comai

Time:

Monday	8:15-11:15
Friday	15:15-17:15

Textbooks

IN ITALIAN:

P. Atzeni, S. Ceri, P. Fraternali, S. Paraboschi, R. Torlone
"*Basi di dati: Architetture e linee di evoluzione*" (2003)

P. Atzeni, S. Ceri, P. Fraternali, S. Paraboschi, R. Torlone
"*Basi di dati*" (2018)

IN ENGLISH:

P. Atzeni, S. Ceri, S. Paraboschi, R. Torlone
"*Database systems*" - McGraw-Hill (1999)

NOW DOWNLOADABLE (http://dbbook.dia.uniroma3.it/)

Compared to the last Italian version some chapters are missing

More bibliobraphy on specific topics (e.g., JPA) listed in the official program page of the course

Teaching material

- Material is available on the Beep portal

http://beep.metid.polimi.it

under "**Data bases 2**"

- Materials include slides of the lectures, exercise sessions, forum, etc.

- If your study plan has not been approved yet, you can subscribe to the course. Please specify the motivation!

App for iOS

- App to solve/check the classification of the schedules (VSR, CSR, 2PL etc.).
- Available on the App Store

"DBSA" (Data Base Schedule Analyzer)

https://itunes.apple.com/us/app/data-base-schedule-analyzer/id619821068?l=it&ls=1&mt=8

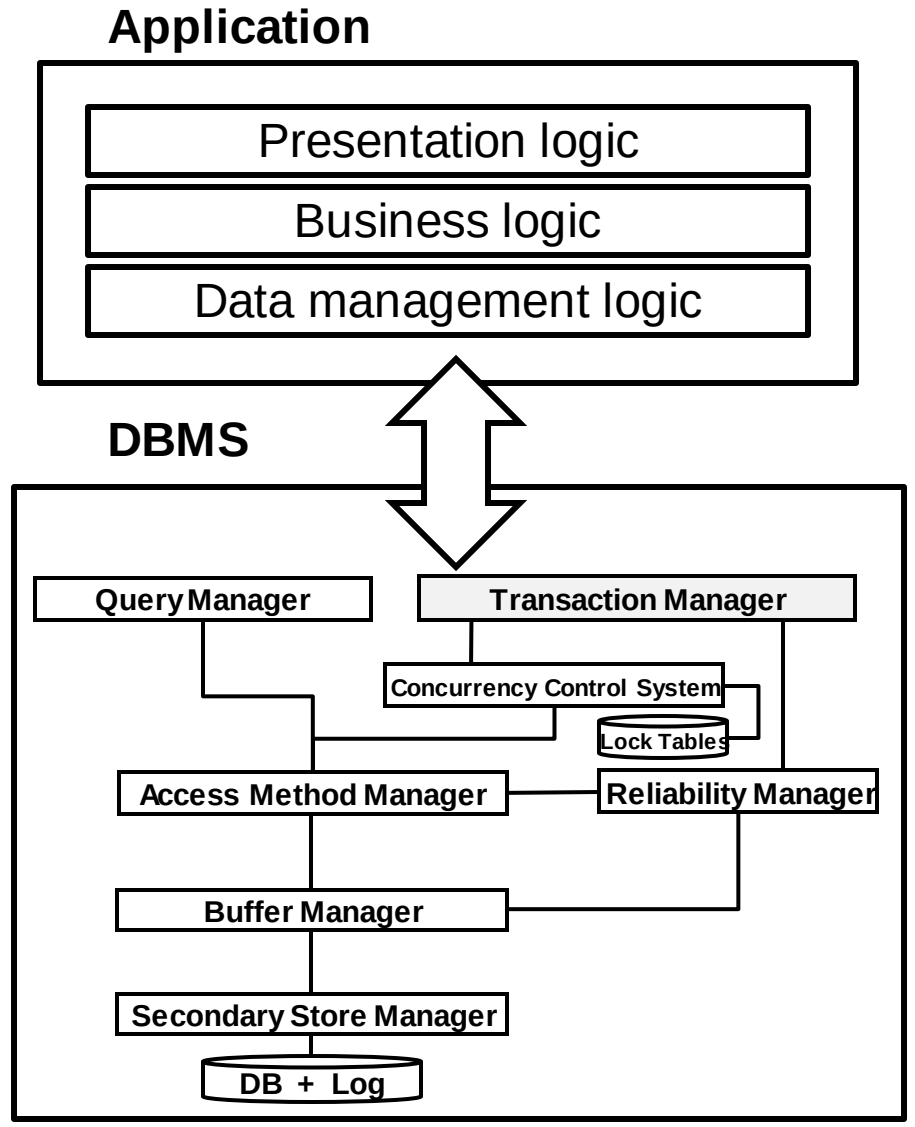
- Of course you cannot use it during the **exams: mobile devices** will be **forbidden** also to check the time.

Prerequisites

- Basics of Database systems
  - Relational model
  - SQL (SQL-92)
  - Relational algebra
  - Object oriented design with UML and programming with Java
  - Recommended for those who want to do the optional project: Web programming with Java EE
    - prerequisite materials in English can be found in Beep, course Tecnologie Informatiche per il Web // Web Technologies, lessons and exercises on Servlet programming.
  - **BOOK: http://dbbook.dia.uniroma3.it/**

What you will learn from the course

- Opening the database box
  - Transaction management
  - Concurrency control
  - Reliability manager
  - Physical data management and query optimization
  - Data distribution
- Opening the application box
  - Mapping objects to tables
  - Mapping methods to transactions
  - Managing the alignment between object state in main memory and data state in secondary memory



DataBase Management System — DBMS

- A system (software product) capable of managing data collections that are:
  - **Large** ((much) larger than the central memory available on the computers that run the software)
  - **Persistent** (with a lifetime which is independent of single executions of the programs that access them)
  - **Shared** (in use by several applications at a time)
  - **Reliable** (i.e. ensuring tolerance to hardware and software failures)
  - **Privacy respecting** (by disciplining and controlling all accesses).

Technology of DBMSs – questions

- Transaction management
  - How do I ensure that a set of updates is executed "all or nothing"? How do I undo the whole set if one update fails? → Acid properties
- Concurrency control
  - How do I ensure that my transaction executes "correctly" in presence of other transactions that update the same data? → CC theory, pessimistic and optimistic locking
- Reliability control
  - How do I avoid data losses in presence of failures? → Log and recovery protocols
- Distributed architectures
  - How can I distribute data and process queries across multiple hosts? → Fragmentation, transparency, distributed recovery
- Buffer and secondary memory management
  - How and when are data moved from main to secondary storage? → paging and caching
- Physical data structures and access structures
  - What are the low level data structures that store data? → sequential, hash-based and tree-based structures
- Query management
  - How can data organization be exploited to speed up queries? → cost-based optimization

Database-Application integration: questions

- How do I address the impedance mismatch between database and application models (Table vs Class, Tuple vs Object, Pointers vs Relations and Foreign Keys)?
  → code level procedure, Object Relational Mapping
- How do I make my application communicate with the database?
  → call level interface, ODBC-JDBC, JPA Persistence Provider
- How can I align the state of an object and the state of the persistent data that correspond to it?
  - How do I map changes to objects to changes to persistent data?
    → JPA managed entities
- How can I ensure that application functions (e.g., object methods) execute with the proper "all or nothing" semantics in the database? How do I map method calls to database transactions? → JDBC transactions, JTA transactions, transaction scoped persistence contexts

Related topics addressed in other courses

- 054306 - UNSTRUCTURED AND STREAMING DATA ENGINEERING
  - NOSQL systems
    - graph, column, document, key-value based storage; persistent and volatile solutions
  - BASE transaction models
  - Stream data storage: time series, data streams and events data bases and processing pipelines

DB Evolution

Since the 70's: relational databases + SQL
Some revolutions in the 90's:

- SQL'92
- SQL'99 (triggers, object-oriented features)

And more recently:

- SQL:2003 (XML-related features)
- SQL:2006 (XQuery)
- SQL:2011 (Temporal DB)
- SQL:2016 (row pattern matching, JSON)

- Since 2005: NoSQL DBMS (no standard!)
- A single application may involve different kinds of data → diverse data models and query languages for a single application

Popularity of the models

Rank Aug 2020	Rank Jul 2020	Rank Aug 2019	DBMS	Database Model	Score Aug 2020	Score Jul 2020	Score Aug 2019
1.	1.	1.	Oracle	Relational, Multi-model	1355.16	+14.90	+15.68
2.	2.	2.	MySQL	Relational, Multi-model	1261.57	-6.93	+7.89
3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	1075.87	+16.15	-17.30
4.	4.	4.	PostgreSQL	Relational, Multi-model	536.77	+9.76	+55.43
5.	5.	5.	MongoDB	Document, Multi-model	443.56	+0.08	+38.99
6.	6.	6.	IBM Db2	Relational, Multi-model	162.45	-0.72	-10.50
7.	↑ 8.	↑ 8.	Redis	Key-value, Multi-model	152.87	+2.83	+8.79
8.	↓ 7.	↓ 7.	Elasticsearch	Search engine, Multi-model	152.32	+0.73	+3.23
9.	9.	↑ 11.	SQLite	Relational	126.82	-0.64	+4.10
10.	↑ 11.	↓ 9.	Microsoft Access	Relational	119.86	+3.32	-15.47
11.	↓ 10.	↓ 10.	Cassandra	Wide column	119.84	-1.25	-5.37
12.	12.	↑ 13.	MariaDB	Relational, Multi-model	90.92	-0.21	+5.96
13.	13.	↓ 12.	Splunk	Search engine	89.91	+1.64	+4.03
14.	↑ 15.	↑ 15.	Teradata	Relational, Multi-model	76.78	+0.81	+0.14
15.	↓ 14.	↓ 14.	Hive	Relational	75.29	-1.14	-6.51
16.	16.	↑ 18.	Amazon DynamoDB	Multi-model	64.75	+0.17	+8.18
17.	↑ 18.	↑ 25.	Microsoft Azure SQL Database	Relational, Multi-model	56.85	+4.22	+28.85
18.	↓ 17.	↑ 20.	SAP Adaptive Server	Relational	53.96	+0.09	-1.90
19.	↑ 20.	↑ 21.	SAP HANA	Relational, Multi-model	53.12	+1.78	-2.31
20.	↓ 19.	↓ 16.	Solr	Search engine	51.69	+0.05	-7.43
21.	↑ 22.	↑ 22.	Neo4j	Graph	50.18	+1.26	+1.79
22.	↑ 23.	↓ 19.	HBase	Wide column	49.11	+0.45	-7.42

Ranking from http://db-engines.com/en/ranking

Exam

- The exam consists in a written verification covering all the topics of the course
  - Exercises on the whole program, possibly with related theoretical questions.
  - 4 main topics:
    - DB architecture/technologies (concurrency control etc.)
    - Trigger
    - ORM and JPA
    - Physical DB/Query optimization
- An **optional** project is offered, which requires programming an application with Java EE and JPA

Exam rules

- During the exam:
  - No books, notes, electronic devices are allowed
  - Cheating policies: it is forbidden to communicate with other students. Students talking to each other are asked to leave the classroom (both!)
  - No oral exams will be done

- **For the optional project verification**: an individual oral colloquium will be scheduled where the student makes a demo of the project and a code walkthrough is performed

Exam rules

- Positive marks can be **rejected**
  - This can be done through the online system; usually there are 5 days after the insertion of the mark into the system to reject it
  - REMARK: when the exam is repeated (it is sufficient to sit down and see the text of the exam), the **previous mark is lost**!

Academic calendar and DB2 exams

- 1st semester: ends before Christmas

  1st examination session: 2 DB2 exams
  - Between January -February

  2nd examination session: 2 DB2 exams
  - Between June -- July

  3rd examination session: 1 DB2 exam
  - August – September

- **Colloquium dates for the optional project defined on individual basis with the teachers.**
  - Colloquium dates are within the exam session where the student has done the written part