Introduction

Prof. Alfons Kemper &
Alexander van Renen
Chair for Database Systems (I3)
www.db.in.tum.de

TU München / Garching
[kemper,renen]@in.tum.de

Lecture

- Web page of the lecture: see TUMonline

  www-db.in.tum.de/teaching/ws1819/DBSandere

- IN4714:
  - Part of the module **Geodatabases** (BV470015)
  - Duration: 2V SWS
  - Credits: 2 ECTS

Schedule

- 2 hours weekly
- Wednesdays, 4.45 – 6.15 p.m.
- Exam (closed book): Date not fixed yet
- BV470015: 60 minutes
- IN4714: 40 minutes
- Minimum number of points: 50%

Teaching

- Questions during class are very welcome
- Reading material for preparation
- Beforehand distributed / embedded excercises
- Discussion of problems / solutions on the whiteboard

→ interactive class !!

Overview

- Database Design
  - E/R-Modeling
  - UML-Modeling
- Relational Data Model
- Relational Query Language SQL
- Data Integrity

Overview (cont.)

- Physical Data Organization
  - B-Trees
  - Hashing
- Query Execution
- Transaction Management
- (Main Memory Databases,
  NoSQL Databases, Data Warehouses, …)

→ Preparation for Geodatabases,
Andreas Donaubauer, starting December 2018

Material used

Slides of Prof. Kemper:

www-db.in.tum.de/teaching/bookDBMSeinf

and Prof. Neumann:

www-db.in.tum.de/teaching/ws1415/grundlagen

(both in German)

Literature (in German)

Alfons Kemper und André Eickler

Datenbanksysteme: Eine Einführung

10. Auflage (2015)

(older Editions are also ok)

Oldenbourg Verlag, München

(~ 50 Euros)

www-db.in.tum.de/teaching/bookDBMSeinf

Associated Workbook

Alfons Kemper und Martin Wimmer
Übungsbuch Datenbanksysteme
3. Auflage (2011)
(older Editions are also ok)
Oldenbourg Verlag, München
(~ 35 Euros)

Additional Material

www-db.in.tum.de/teaching/bookDBMSeinf

- Slides
- Videos of lectures
- Data to build own databases
- Programming examples for
  - IBM DB2
  - Oracle
  - MS SQL Server

Additional Material

tools.db.in.tum.de

- Many useful database tools

http://hyper-db.com/interface.html

- A SQL webinterface based on HyPer

Literature (in English)

A. Silberschatz, H. F. Korth und S. Sudarshan

Database System Concepts, 6th edition, McGraw-Hill, 2010. codex.cs.yale.edu/avi/db-book/db6/slide-dir/

R. Elmasri, S.B. Navathe

Fundamentals of Database Systems, 6th edition, Addison-Wesley, 2010. (also available in German)

R. Ramakrishnan, J. Gehrke

Database Management Systems, 3th edition, 2003. http://pages.cs.wisc.edu/~dbbook/

Literature (cont.), MOOCS

J.D. Ullmann, J. Widom

A First Course in Database Systems, Prentice Hall, 3rd edition, 2007. infolab.stanford.edu/~ullman/fcdb.html

MOOCS

- Self paced mini courses, Stanford class.stanford.edu/courses/DB/2014/SelfPaced/about
- Datenmanagement mit SQL, HPI open.hpi.de/courses/sql (in German)

MOOCS (cont.), Lectures online

- Informationssysteme/
  Einführung in Datenbanksysteme, Uni Saarland
  infosys.uni-saarland.de/datenbankenlernen/
  (partly in German)

- Lecture online
  ETHZ, D. Kossmann, spring 2014:
  http://www.video.ethz.ch/lectures/d-infk.html
  English slides

Terms

- What is a database system (DBS)?

System to store and manage data

- Why not use a traditional file system?

Reliability and scalability only achievable with high effort

Examples

Traditional application areas:

- business data
- accounting
- administration

...

Nowadays a lot broader:

- scientific / medical data
- data mining
- geographical information systems
- web search

...

Examples (cont.)

Databases are the back of many applications:

- web search with Google, Yahoo, ...
- inquiries to Amazon, EBay, ...
- posts in Facebook, Twitter, …

Many varieties (DBS/Information Retrieval, centralized/decentralized, replicated, etc.)

Databases are used whenever

- data is very precoius (→ reliability)
- amount of data is very big (→ scalability)

Examples (cont.)

The big commercial database systems:

- Oracle
- IBM DB2
- Microsoft SQL Server

Some open source database systems:

- PostgreSQL
- MySQL
- MonetDB

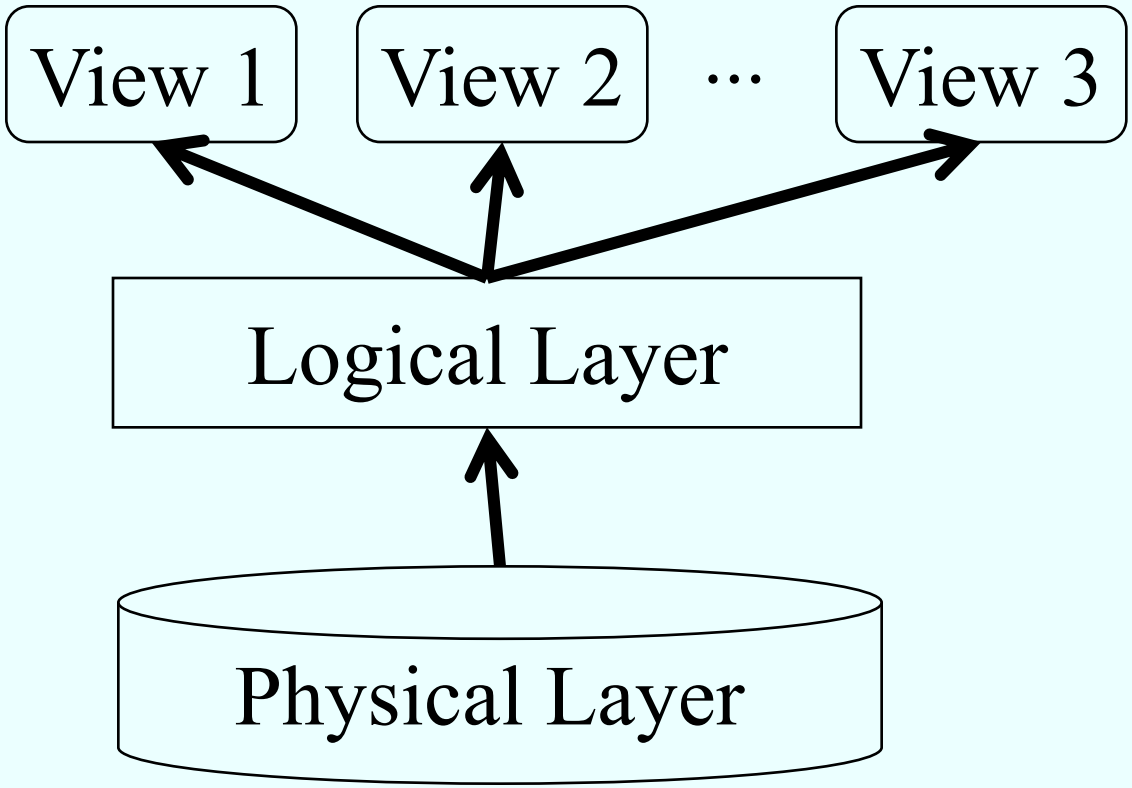
Many more, some very specialized (XML, object oriented, data streams, …)

Why use a database system?

- Avoid redundancy and inconsistency
- Rich (declarative) access to the data
- Security and privacy issues
- Synchronize concurrent data access
- Avoid loss of data
- Recovery after system failures
- Efficiency and scalability

→ Concentrate on your business logic

Abstract layers of a database system



Abstract layers of a database system (cont.)

View:
describes how a user / program sees the data

Logical layer:
describes how the data is structured

Physical layer:
describes how the data is stored

Abstract layers of a database system (cont.)

DBS decouples applications from the structure and storage of the data:

- **Logical data independency**
  (simple) changes at the logical layer have no influence on the applications
- **Physical data independency**
  changes at the physical layer have no influence on the applications

Implemented in almost all modern database systems

Properties of database systems

Data integrity (consistency)

- Data processing within an application has constraints

→ DBS obeys defined rules and protects automatically from:

- User errors
- Programming errors

Properties of DBSs (cont.)

Declarative query language

- User determines ***which*** data should be retrieved . . .
  . . . and ***not how***

→ Less error-prone (when querying the data / developing applications)

→ No knowledge about the interior layers of the DBS necessary

Properties of DBSs (cont.)

Sophisticated access rights

- Every user can get different rights on the database

→ DBS provides a variety of access control mechanisms to enable security and privacy

Properties of DBSs (cont.)

Multi user concurrency

- If you allow several users at a time to update the data without any control you run into big problems

→ DBS allows concurrent access and avoids side effects

Properties of DBSs (cont.)

Error handling

- DBS can restore its state consistently in case of a system failure

→ Therefor log files are held and managed by the DBS

Properties of DBSs (cont.)

Efficiency and scalability

- DBSs are designed for efficiently handling very large data volumes and a very high number of users

→ In DBSs techniques for scaling with ever higher data volumes are integrated

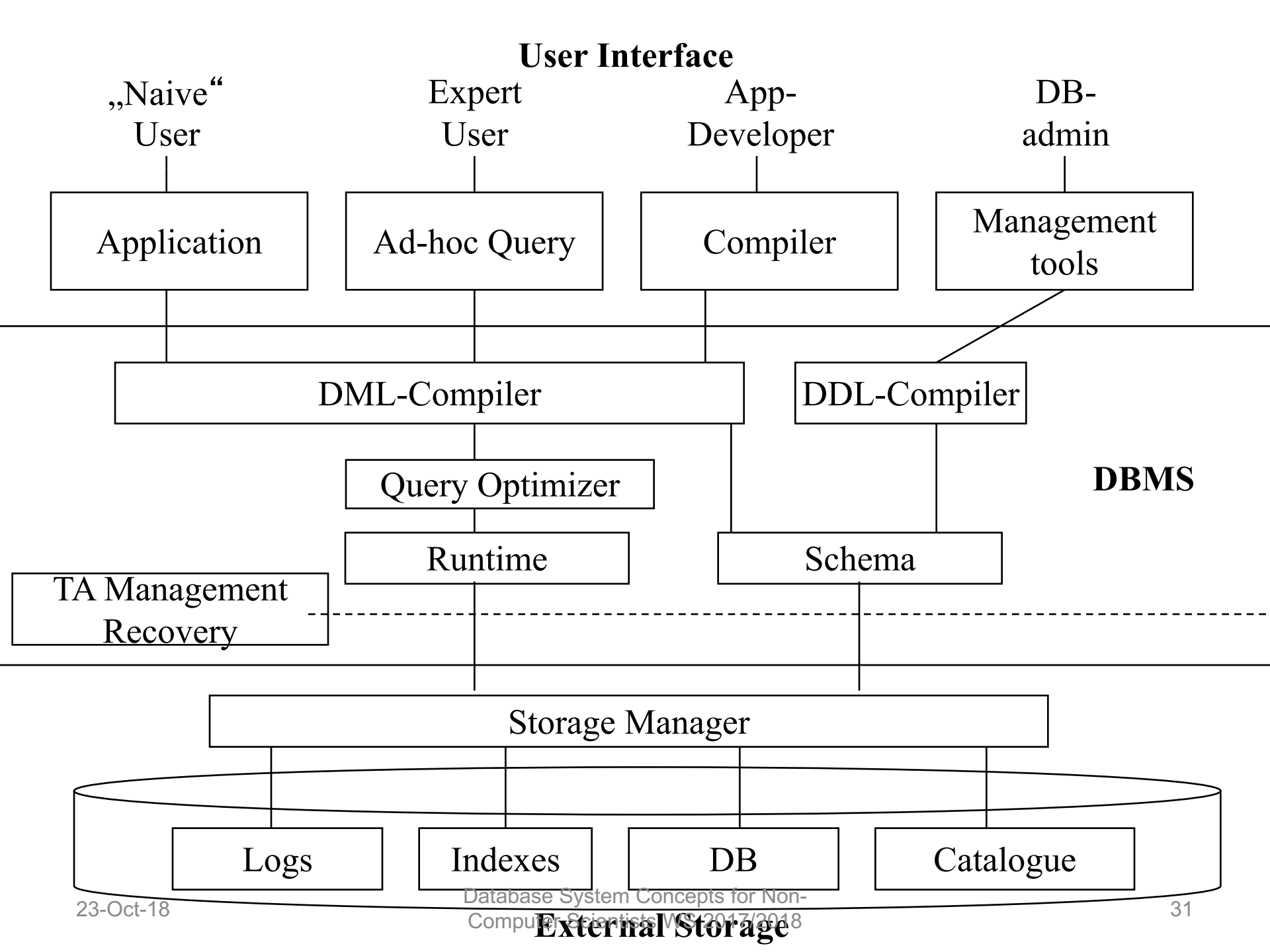
typically: 100 GB (Gigabyte) – transactional Daten (even express versions)
up to EB (Exabyte) maximum data size

Properties of DBSs (résumé)

- Data integrity
- Declarative query language
- Access rights
- Concurrency control
- Error handling
- Efficiency and scalability

Architecture & Components of a Database System

- Layered architecture
  - User Interface
  - DBMS
  - External Storage


User Interface
„Naive“ User
Expert User
App-Developer
DB-admin
Application
Ad-hoc Query
Compiler
Management tools
DML-Compiler
DDL-Compiler
Query Optimizer
DBMS
Runtime
Schema
TA Management Recovery
Storage Manager
Logs
Indexes
DB
Catalogue
External Storage

Next: Data Modeling

Manual Modeling