University of Zurich[UZH]

GEO 874 | HS25
University of Zürich

Introduction to Databases

Esra Suel
Dept. of Geography, University of Zürich
Rolf Meile
Eidg. Forschungsanstalt für Wald, Schnee und Landschaft (WSL)

Course materials have been adapted from prior versions developed by Dr. Zhiyong Zhou, Dr. Cheng Fu, and Dr. Haosheng Huang. Credit and thanks for their contributions.

Welcome

Thank you for being here at 8am on Friday!

The GEO 874-Team: Lecturers

Esra Suel, Professor of Urban Analytics, GIUZ
CV: PhD Transportation and Urban Systems, Civil and Environmental Engineering (Imperial College London)
Interests: urban analytics, transportation/mobility, housing, built environment
Contact: esra.suel@geo.uzh.ch,Y25 L90

Rolf Meile, Database- and GIS-Specialist, WSL
CV: MSc. Biology (UZH), Monitoring Institute (St. Gallen), Cityline AG ZH
Interests: spatio-temporal aspects of data modelling, data quality management, GIS-CAD, molecular cooking
Contact: rolf.meile@wsl.ch

A course with history – and a fresh start

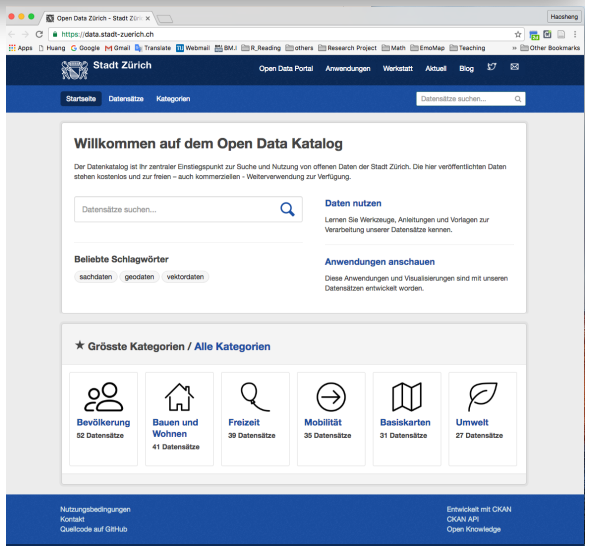
- First time I'm teaching databases & at UZH.
- But: this course has a long tradition at GIUZ
- Lucky to co-teach with Rolf (16 years' experience!)
- Consistently popular and useful for careers.

My approach: learn together, feedback welcome

Goal: supportive, mistakes welcome, bring material alive

Real world...

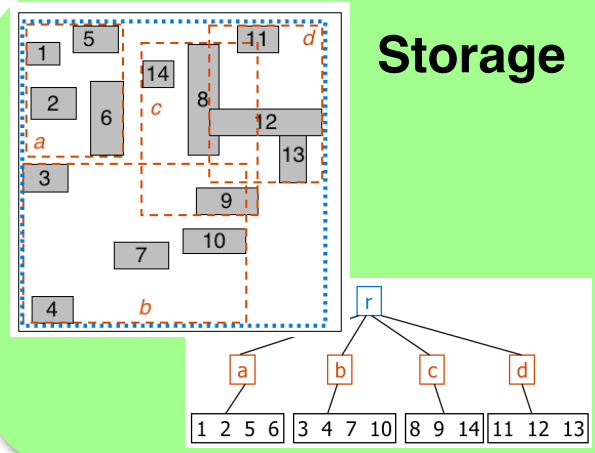
- **The Swiss National Forest Inventory**
  - 80'000 trees measured along with 30 attributes
  - Stored in a database
  - Analysis of the thematic and spatial characteristics of the forest
- **Open Data Zürich**
  - Open government Data
  - > 250 datasets
  - Different topics: population, environment, mobility, ...
  - Spatial data and non-spatial data

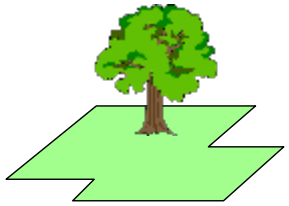


➲ http://www.lfi.ch/lfi/lfi_film.php
➲ https://data.stadt-zuerich.ch/

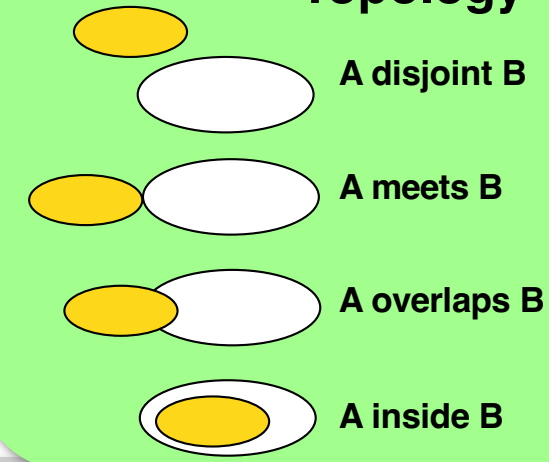
Laying the foundations

Storage

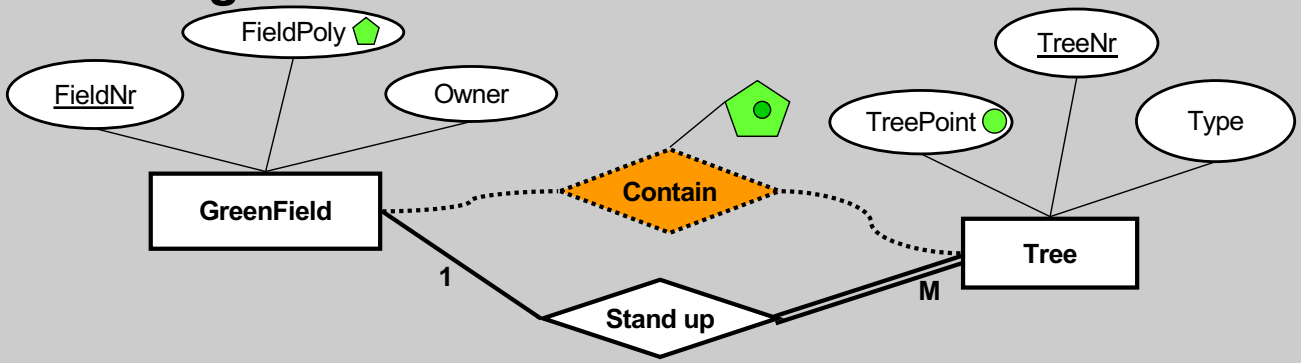


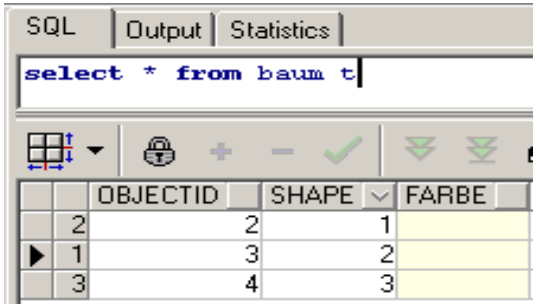


Topology



Modelling





GEO 875:
Spatial databases

GEO 874:
Introduction
to databases

Schedule

L1	**Introduction to databases: Definitions and Concepts**	"What are databases and what are their advantages?"
L2-L3	**SQL (Structured Query Language)**	"How can we define, modify and query a DB?" **SQL Assignment**
L3-L5	**Database design**	"What are the THINGS that should be stored in a DB, and what are their attributes and relationships?" Conceptual design → Logical design → Physical design
L6	**Advanced topics, feedback**	Distributed databases and NoSQL database, Big data
L7	**Written exam (1 hour)** 31.10. 2025 (Fri)	

Feedback and Evaluation

- 📝 **Informal check-in** (Week 6)
  - Quick feedback in class
  - Adjustments during the semester

- ✅ **Formal evaluation** (UZH-level)
  - Anonymous
  - Helps improve the course for future years

- 🙌 **Your voice matters**
  - First time I'm teaching this course
  - I've kept it close to the original popular format
  - Your feedback will help me build and improve for the next years

Learning objectives GEO 874

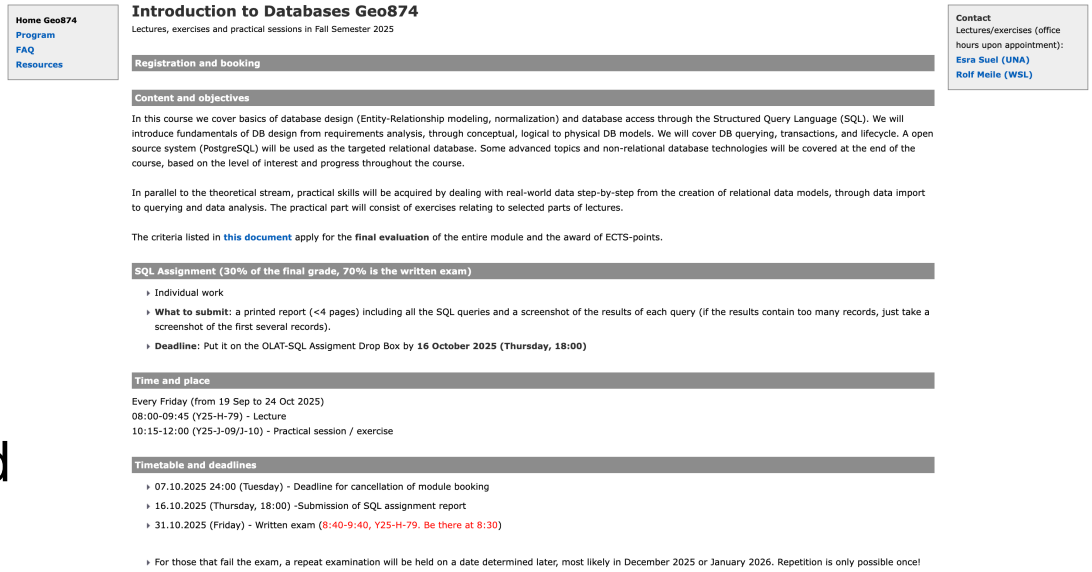
- ✓ Understand why we need databases, and why they are deployed in large and complex projects
- ✓ Understand the fundamentals of (relational) DBMS and apply this knowledge in the scope of non-spatial data
- ✓ You will master the skills of database design and will be able to apply them in a small project
- ✓ You will be able to use SQL (Structured Query Language) to define, modify and query a DB
- ✓ You will understand that there are other ways to store and manage data in a database than the relational model.

Course Website

http://www.geo.uzh.ch/microsite/geo874/

Online contents:

- Lecture notes
  - Week by week
  - Usually on Thursday noon
- Exercise handouts
- Evaluation criteria
- FAQ (Problems, Tips and Tricks)
  1. FAQ-site, Google
  2. Colleagues
  3. Finally, email us
- Resources



Home Geo874
Program
FAQ
Resources

Introduction to Databases Geo874

Lectures, exercises and practical sessions in Fall Semester 2025

Contact
Lectures/exercises (office hours upon appointment):
Esra Suel (UNA)
Rolf Meile (WSL)

Registration and booking

Content and objectives

In this course we cover basics of database design (Entity-Relationship modeling, normalization) and database access through the Structured Query Language (SQL). We will introduce fundamentals of DB design from requirements analysis, through conceptual, logical to physical DB models. We will cover DB querying, transactions, and lifecycle. A open source system (PostgreSQL) will be used as the targeted relational database. Some advanced topics and non-relational database technologies will be covered at the end of the course, based on the level of interest and progress throughout the course.

In parallel to the theoretical stream, practical skills will be acquired by dealing with real-world data step-by-step from the creation of relational data models, through data import to querying and data analysis. The practical part will consist of exercises relating to selected parts of lectures.

The criteria listed in this document apply for the **final evaluation** of the entire module and the award of ECTS-points.

SQL Assignment (30% of the final grade, 70% is the written exam)

- Individual work
- **What to submit**: a printed report (<4 pages) including all the SQL queries and a screenshot of the results of each query (if the results contain too many records, just take a screenshot of the first several records).
- **Deadline**: Put it on the OLAT-SQL Assigment Drop Box by **16 October 2025 (Thursday, 18:00)**

Time and place

Every Friday (from 19 Sep to 24 Oct 2025)
08:00-09:45 (Y25-H-79) - Lecture
10:15-12:00 (Y25-J-09/J-10) - Practical session / exercise

Timetable and deadlines

- 07.10.2025 24:00 (Tuesday) - Deadline for cancellation of module booking
- 16.10.2025 (Thursday, 18:00) -Submission of SQL assignment report
- 31.10.2025 (Friday) - Written exam (8:40-9:40, Y25-H-79. Be there at 8:30)

- For those that fail the exam, a repeat examination will be held on a date determined later, most likely in December 2025 or January 2026. Repetition is only possible once!

OLAT

25HS GEO874.1 Introduction to Databases, Exercises

Primarily for:

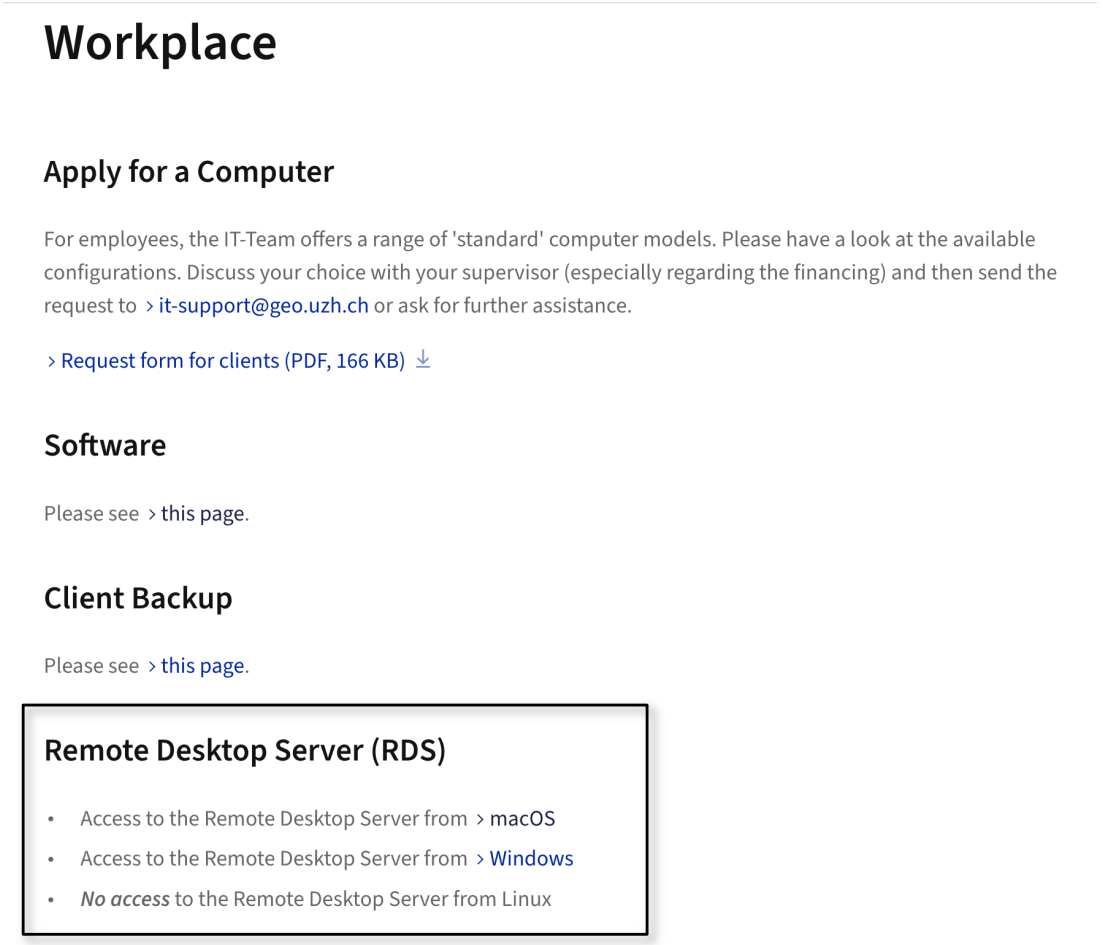
- 📣 Announcements / group messages
- 💻 SQL assignment

Note:

- The course website is embedded within OLAT.
- We will **NOT** use *25HS GEO874.2 Introduction to Databases, Exercises*

Remote Desktop

- https://www.geo.uzh.ch/en/services/it-services/workplace.html
- You would need this when you want to access the “Remote Desktop Server” from your own computer



Workplace

Apply for a Computer

For employees, the IT-Team offers a range of 'standard' computer models. Please have a look at the available configurations. Discuss your choice with your supervisor (especially regarding the financing) and then send the request to › it-support@geo.uzh.ch or ask for further assistance.

› Request form for clients (PDF, 166 KB)

Software

Please see › this page.

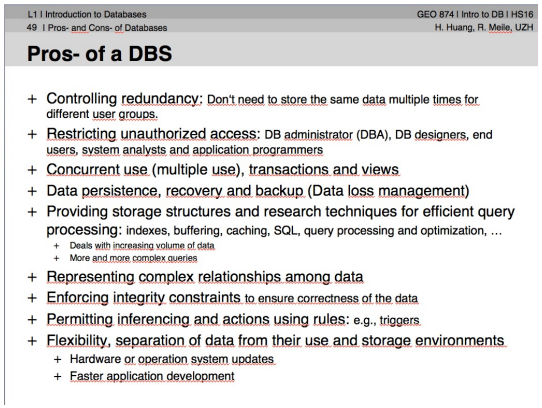
Client Backup

Please see › this page.

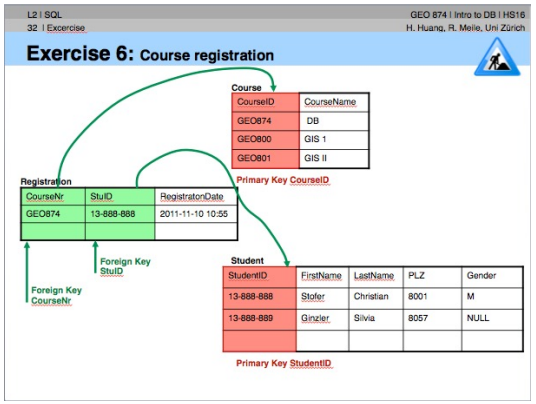
Remote Desktop Server (RDS)

- Access to the Remote Desktop Server from › macOS
- Access to the Remote Desktop Server from › Windows
- *No access* to the Remote Desktop Server from Linux

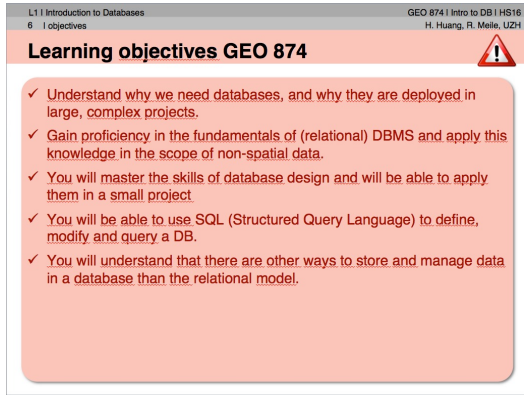
We will use the following colour code



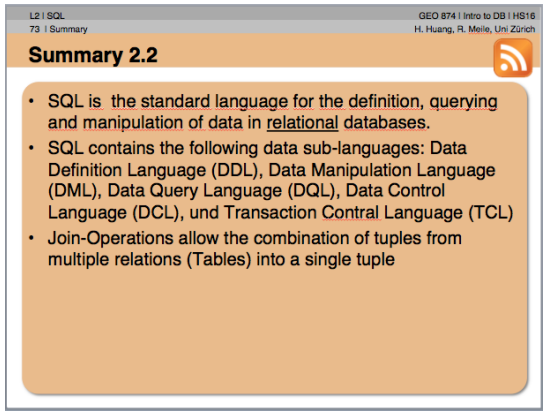
Main contents (just basics, take notes!)



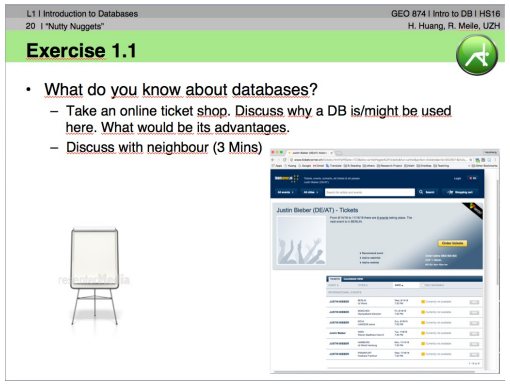
On your own…



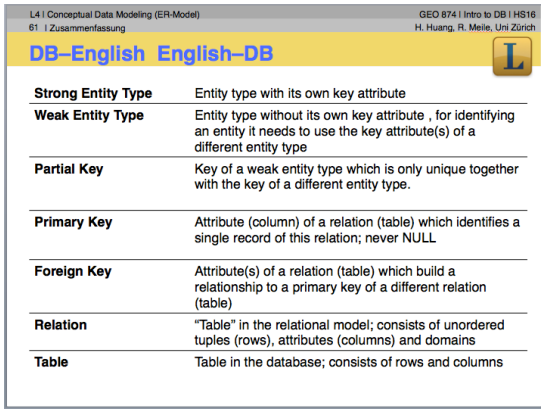
Learning objectives



Summary



Exercises in the class



Strong Entity Type	Entity type with its own key attribute
Weak Entity Type	Entity type without its own key attribute , for identifying an entity it needs to use the key attribute(s) of a different entity type
Partial Key	Key of a weak entity type which is only unique together with the key of a different entity type.
Primary Key	Attribute (column) of a relation (table) which identifies a single record of this relation; never NULL
Foreign Key	Attribute(s) of a relation (table) which build a relationship to a primary key of a different relation (table)
Relation	"Table" in the relational model; consists of unordered tuples (rows), attributes (columns) and domains
Table	Table in the database; consists of rows and columns

Terminology, Glossary

Exercises & Practicals (Computer lab)

✏️ Theoretical Exercises

- Modelling with „Pen and Paper“

💻 Practicals

- On computer
- Creating and modifying tables, incl. constraints
- Loading and modifying data
- SQL-queries

Timeplan: every Friday

Lectures will be on-site.

25-H-79:

- 8:00 – 8:45 Lecture
- 9:00 – 9:45 Lecture

📍 On-site attendance is required.

25-J-09 and 25-J-10:

- 10:15 -12:00 Lab, Practicals, e-learning, demo

📍 On-site attendance is required.

At home:

- individual study, ...

📅 Lecture time from **19.09.2025** to **24.10.2025 (6 Weeks)**

📝 Exam: **31.10. 2025**

Deadlines

Date	
Tue 28.09.2025	Cancellation / Stornierungsfrist
Thu 16.10.2025 (18:00)	Submission of SQL assignment report
Fri 31.10.2025	Exam – 8:40-9:40 (tbc)
Third week of Nov 2025	Exam results Repetition only available to those who failed!
tba Dec 2024 or Jan 2025	Repeat exam

Evaluation criteria

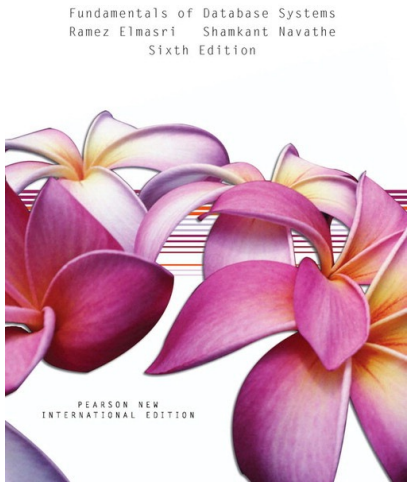
See **GEO 874 — Evaluation criteria for HS 2024**

http://www.geo.uzh.ch/microsite/geo874/scripte/geo874_HS24_criteria.pdf

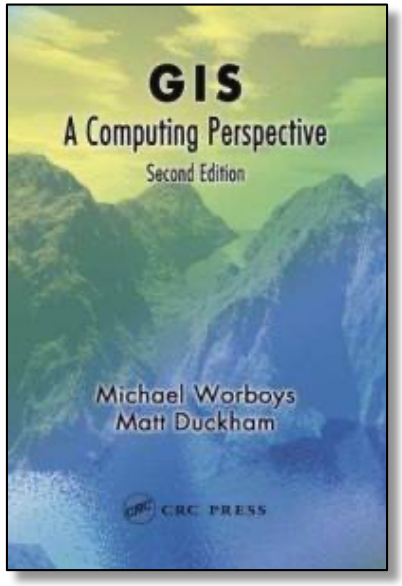
- **Marking of GEO 874**
  - 70% Written Exam
  - 30% SQL Assignment
- **Written Exam**
  - Coverage: all of the Module GEO 874 as per learning objective
  - Type: Written, 1 hr.
  - Marking: Pass necessary (Marked as ≥ 4.0). Max 2 attempts.
- **SQL Assignment**
  - SQL queries on a sample database
  - The grade results from the score for the assignment report.
  - Should be submitted on OLAT
- **Exercises**
  - **A pre-requisite for Exam** (GEO 874.1) is the successful completion of the exercises.

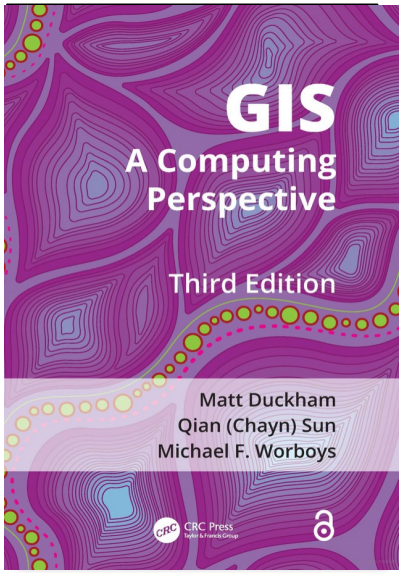
Literature

- Ramez A. Elmasri, Shamkant B. Navathe (2014). Fundamentals of Database Systems, 6th edition, ISBN: 978-1-292-02560-5, Pearson, 1081 pages.
  - There is a newer version 7th edition, which includes NoSQL and big data.
  - Either 6th or 7th edition is fine!



- Michael Worboys, Matt Duckham (2024). GIS – A Computing Perspective, 3rd edition, CRC Press
  - 2nd edition is also used and fine!

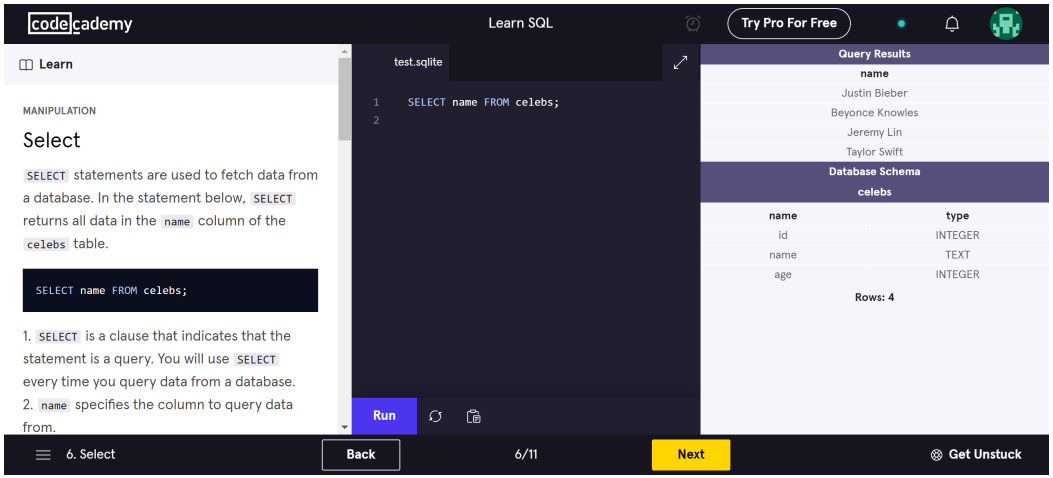




- More literature on the website

Codecademy

- **For interactive learning:** https://www.codecademy.com/learn/learn-sql
- It has introduction and some interactive practices for free
- Do not pay for a pro version



Other resources

When you get stuck…

- 🔍 **Google** – often quickest
- 💬 **Stack Overflow** – lots of SQL answers already there
- 🤖 **ChatGPT (or similar AI tools)** –
  → Useful for explanations & debugging
  → Always double-check answers
  → Don't let it replace your learning!
- 🧑‍💻 **Colleagues / classmates** – discuss & learn together
- 📧 **Email us** – if you still can't solve it

University of Zurich[UZH]

GEO 874 | HS25
University of Zürich

1. Lecture

Introduction to Databases: Definitions and Concepts

Esra Suel
Dept. of Geography, University of Zürich

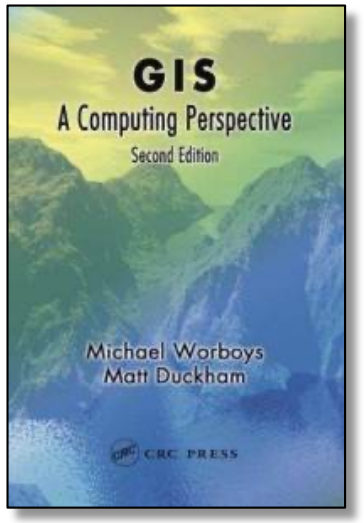
Rolf Meile
Eidg. Forschungsanstalt für Wald, Schnee und Landschaft (WSL)

Learning Objectives 1

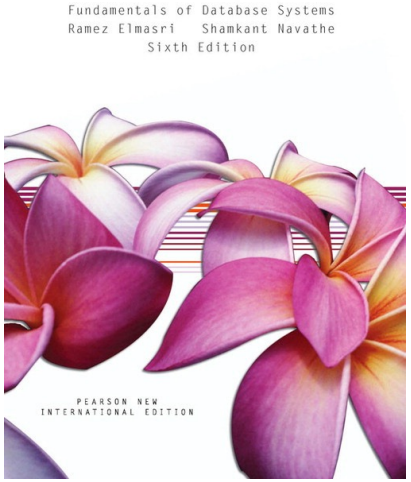
- ✓ Gain a basic understanding of the advantages of Database Management Systems (DBMS) over file-based data handling;
- ✓ Know the main properties of DBMS;
- ✓ Know the basic terminology of DBMS and use them properly and consistently;
- ✓ Understand the fundamentals of the relational model and its application in relational databases;
- ✓ Be able to discuss the pros- and cons- of relational DBMS;
- ✓ Awareness of big „players“ and important application areas of DBMS.

Reading (this Lecture)

- Michael Worboys, Matt Duckham (2004). GIS – A Computing Perspective, 2nd edition, CRC Press, 426 pages.
  Chapter 2.1-2-2: Fundamental Database Concepts, 35-54



- Ramez A. Elmasri, Shamkant B. Navathe (2014). Fundamentals of Database Systems, 6th edition, ISBN: 978-1-292-02560-5, Pearson, 1081 pages.
  Chapter 1: Databases and Database Users,1-26.

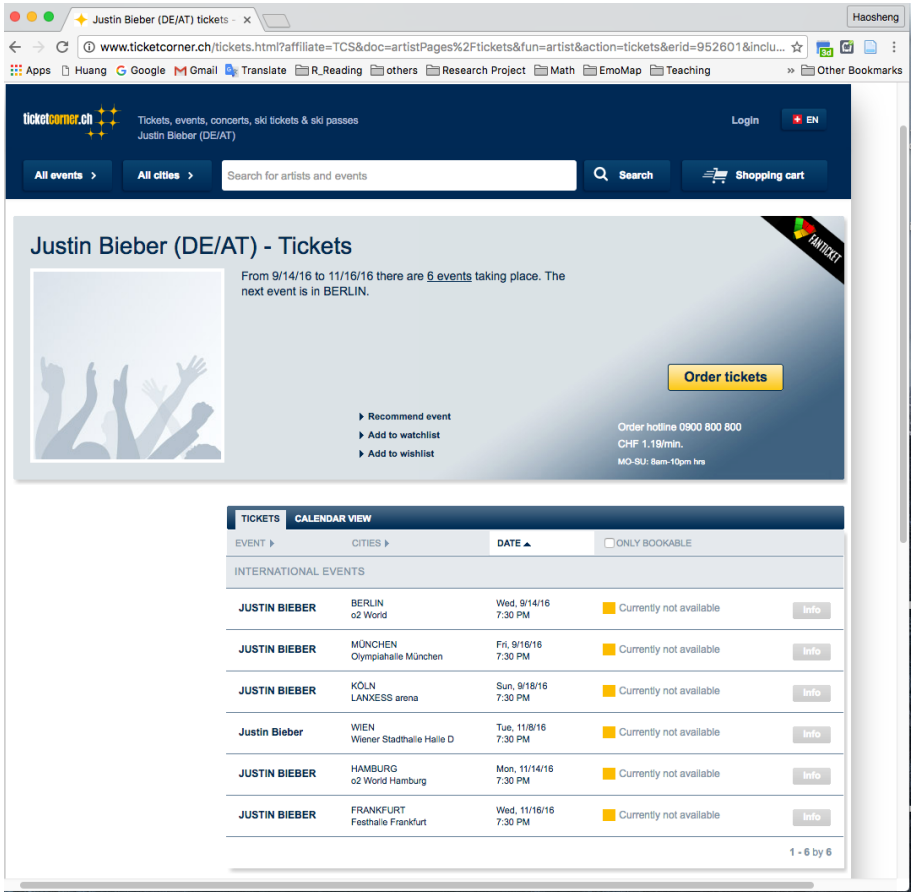


Introduction to Databases

1. **Example: „Nutty Nuggets“**
2. The Database Way...
   - „DB in a nutshell“
   - Database Management System (DBMS)
3. Relational Databases
   - Relational Model
   - Operations on relations
4. Pros- and Cons- of Databases
5. Products, Big Players, Applications

Exercise 1.1

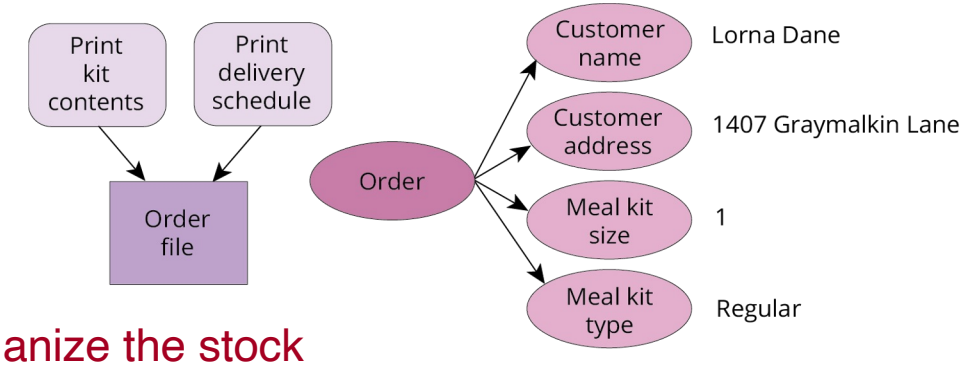
- What do you know about databases?
  - Take an online ticket shop as an example. Discuss why a DB is/might be used here. What would be its advantages (compared to text files or spreadsheets).
  - Discuss with neighbor (3 Mins)



Example: "Nutty Nuggets"

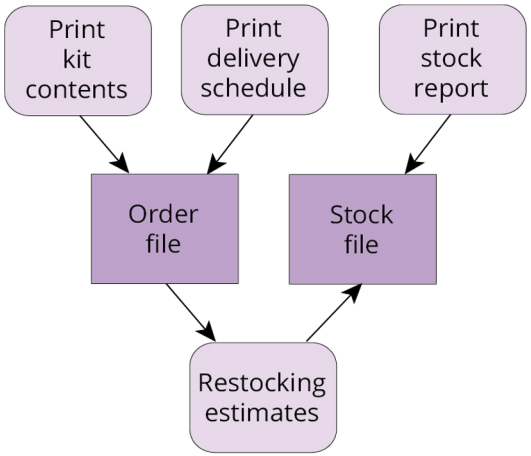
- "Nutty Nuggets": Dinner Kit Company, established in 2018

- **Phase #1**
  - Simple order file
  - A list of orders



We might write a program to organize the stock

- **Phase #2**
  - More
  - + Stock inventory file: raw materials
  - Relationship between stock costs and orders



As time continues, this program will become more complex, offering more functions

➲ Worboys and Duckham (2024) GIS: A Computing Perspective, Third Edition, CRC Press, p. 37-39

"Nutty Nuggets"

- **The file-based system continued to grow**
  - adding new files: suppliers, customer details, ...

- Many problems started to emerge:
  - ***Loss of Integrity***: problems maintaining the structure and currency of the data.

    When customer address was added in wrong format, the programs access the data produced garbled output or crashed.

  - ***Loss of Independence***: Data and the software are tightly coupled. Change of the data structure/storage requires a partial rewriting of many of the software codes.

    When the option to pause orders was added to the order spreadsheet (customers going on vacation) – the restocking program was not updated to incorporate the data associated with this new feature, so a lot of stock went to waste

  - ***Loss of Security***: Any staff member can see or delete personal data of the manager.

    Kitchen staff member working with the stock file also made a secret copy of the orders file, accessing customer names and personal details.

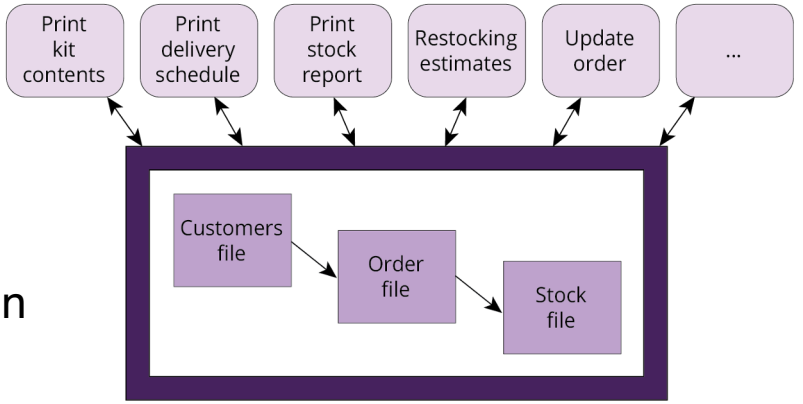
  - …

➲ Worboys and Duckham (2024) GIS: A Computing Perspective, Third Edition, CRC Press, p. 37-39

"Nutty Nuggets"

- **Phase #3, Database**

- Key problems with the previous approach are:
  - Loss of integrity
  - Loss of independence
  - Loss of security



Phase #3, the database, solves these problems

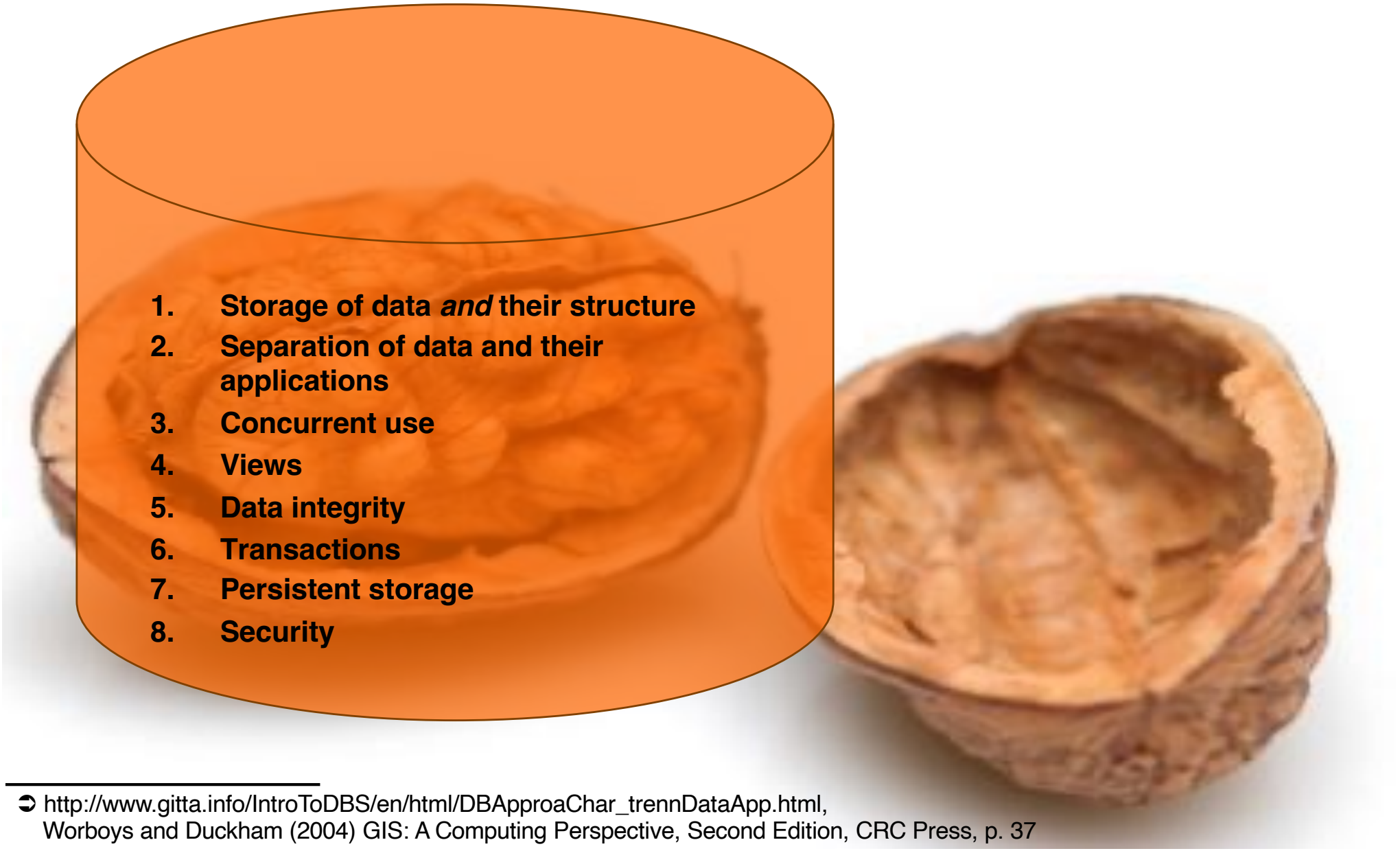
- Integrity: Rules and Constraints (to maintain correctness of the data)
- Independence: Data separated from software applications
- Security: Access control, views
- ...and more!

➲ Worboys and Duckham (2024) GIS: A Computing Perspective, Third Edition, CRC Press, p. 37-39

Introduction to Databases

1. Example: „Nutty Nuggets“
2. **The Database Way ...**
   - „DB in a nutshell“
   - Database Management System (DBMS)
3. Relational Databases
   - Relational Model
   - Operations on relations
4. Pros- and Cons- of Databases
5. Products, Big players, Applications

DB in a nutshell



➲ http://www.gitta.info/IntroToDBS/en/html/DBApproaChar_trennDataApp.html,
Worboys and Duckham (2004) GIS: A Computing Perspective, Second Edition, CRC Press, p. 37

DB in a nutshell: 1. DB stores data *and* their structure

- DB does not only store the data, but also their description.
  - Information about the value types, range, structure …
  - … and their relationships

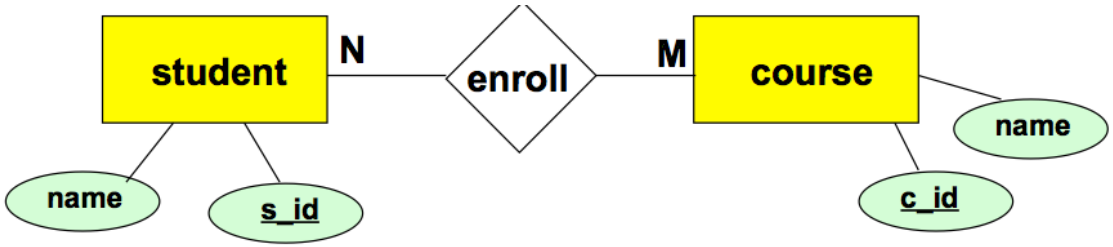


Table: student

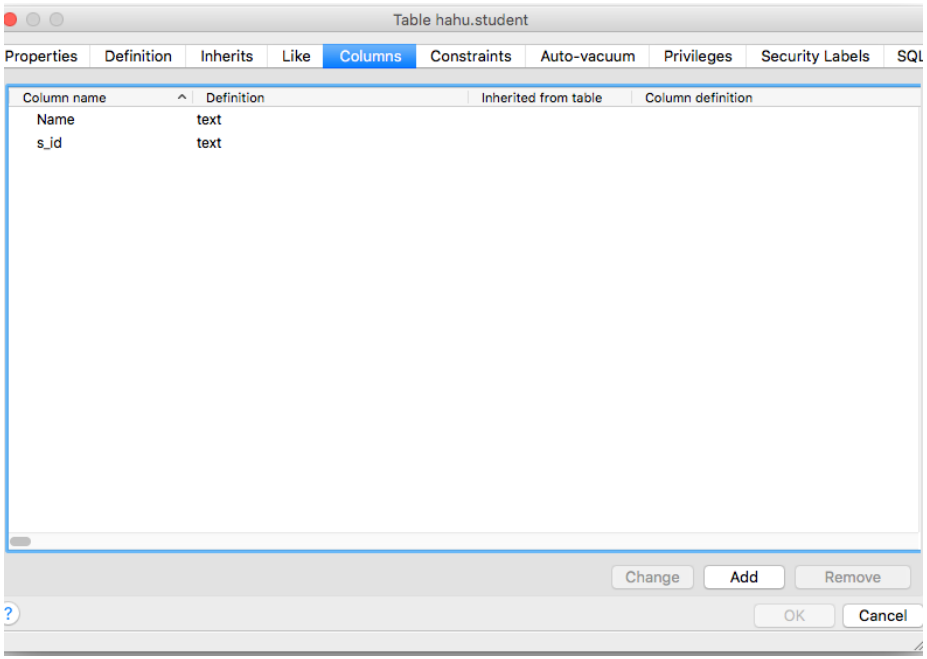
s_id	Name
…	…

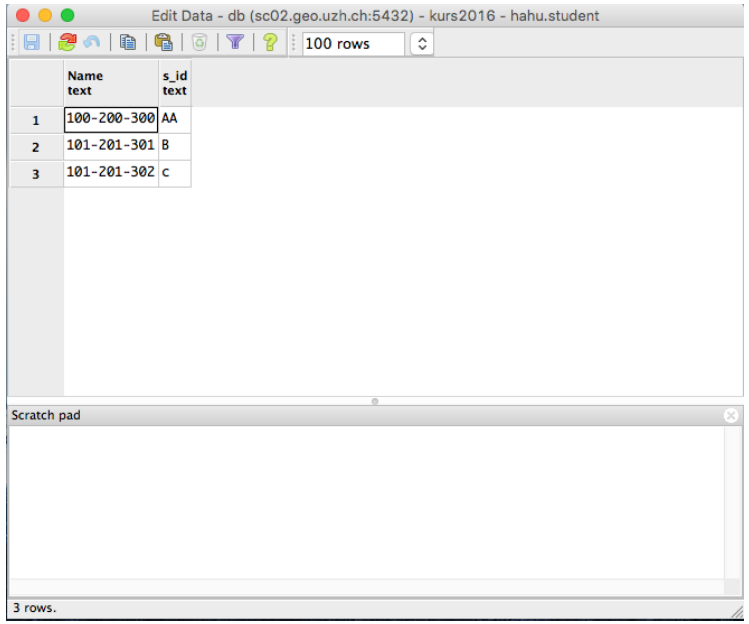
Table: course

c_id	Name
…	…

Table: enroll

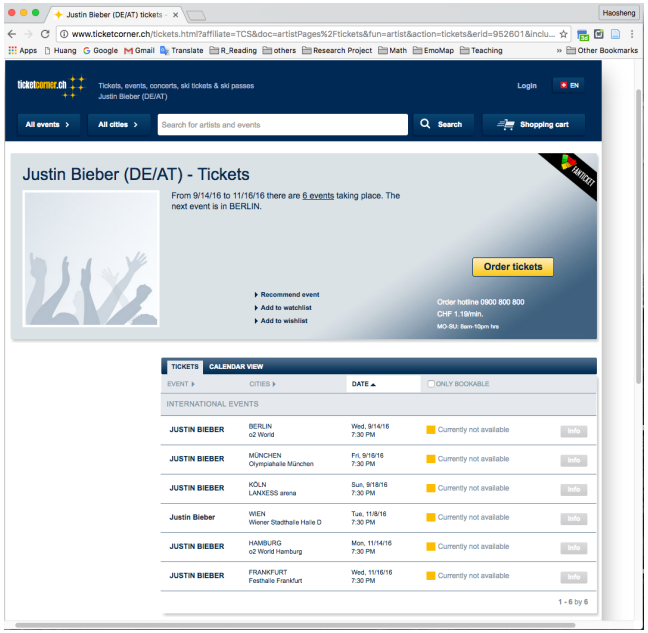
s_id	c_id
…	…

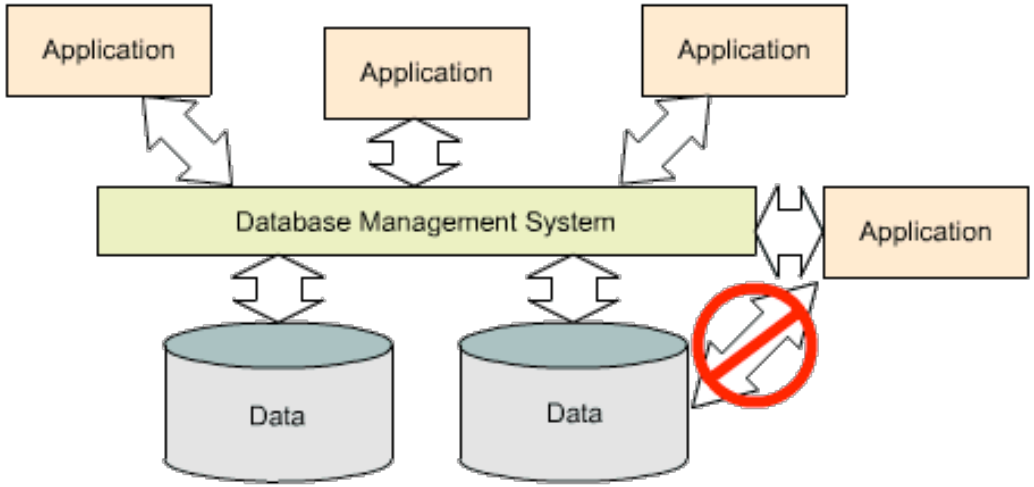




DB in a nutshell: 2. Separation of data and applications

- Applications do not need to know **HOW** the data are physically stored (coding, format, storage location etc.)
- Applications are separated from Data and the access is managed by a DBMS

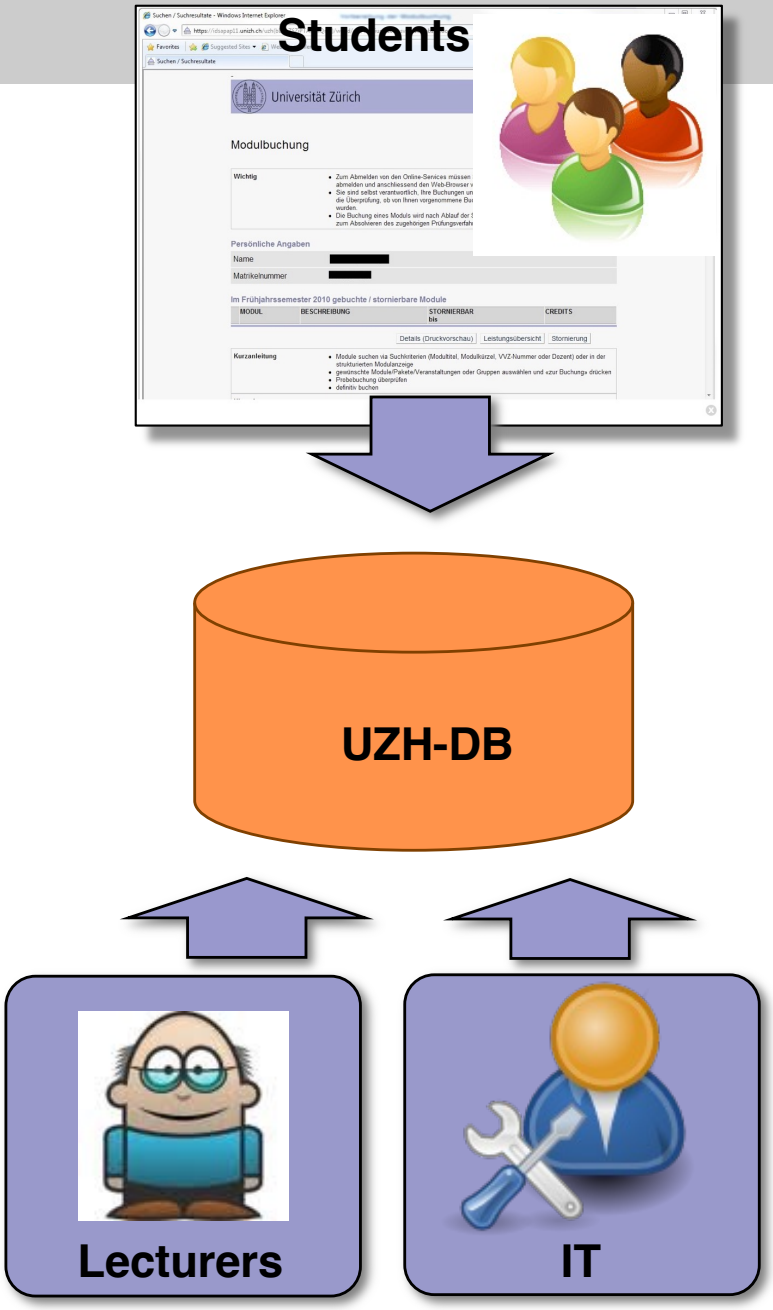




➲ http://www.gitta.info/IntroToDBS/en/html/DBApproaChar_trennDataApp.html

DB in a nutshell: 3. Concurrent Use

- Allows several users to access the DB concurrently (at the same time)
- Responds to queries of diverse users with the same fundamental data



➲ http://www.gitta.info/IntroToDBS/en/html/DBApproaChar_trennDataApp.html

DB in a nutshell: 4. Views

- Different users can have different views of the same data (based on their access rights and needs)

Database Administrator: Whole database	Administration: Statistics about Resits	Classes: Student Lists	Lecturer G5: Class Information

Matriculation	Name	Prename	Address	Institute	Term	Resit	...
99034	Müller	Josef	Oberwil	G	3	N	
99035	Meier	Peter	Regensdorf	G	3	Y	
99036	Keiser	Mirjam	Muttenz	G	3	N	
99037	Muster	Hans	Jägersdorf	G	3	N	
98052	Kunz	Jakob	Bichwil	G	5	N	
98053	Sutter	Ruth	Malhausen	G	5	N	
98054	Weber	Karl	Burgau	G	5	N	
98055	Beck	Karin	Lefendorf	G	5	N	
98056	Schmid	Jürg	Nebelburg	C	5	Y	
98057	Lüthi	Fritz	Sonnwil	C	5	N	

➲ http://www.gitta.info/IntroToDBS/en/html/DBApproaChar_trennDataApp.html

DB in a nutshell: 4. Views

- Different users can have different views of the same data (based on their access rights and needs)

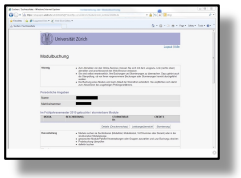
Database Administrator: Whole database | Administration: Statistics about Resits | Classes: Student Lists | Lecturer G5: Class Information

				Institute	Term	Resit	
				G	3	N	
				G	3	Y	
				G	3	N	
				G	3	N	
				G	5	N	
				G	5	N	
				G	5	N	
				G	5	N	
				C	5	Y	
				C	5	N	

➲ http://www.gitta.info/IntroToDBS/en/html/DBApproaChar_trennDataApp.html

DB in a nutshell: 4. Views

- Different users can have different views of the same data (based on their access rights and needs)



Database Administrator: Whole database	Administration: Statistics about Resits	Classes: Student Lists	Lecturer G5: Class Information

	Name	Prename	Address				
	Müller	Josef	Oberwil				
	Meier	Peter	Regensdorf				
	Keiser	Mirjam	Muttenz				
	Muster	Hans	Jägersdorf				
	Kunz	Jakob	Bichwil				
	Sutter	Ruth	Malhausen				
	Weber	Karl	Burgau				
	Beck	Karin	Lefendorf				
	Schmid	Jürg	Nebelburg				
	Lüthi	Fritz	Sonnwil				

➲ http://www.gitta.info/IntroToDBS/en/html/DBApproaChar_trennDataApp.html

DB in a nutshell: 4. Views

- Different users can have different views of the same data (based on their access rights and needs)

Database Administrator: Whole database | Administration: Statistics about Resits | Classes: Student Lists | Lecturer G5: Class Information

Matriculation	Name	Prename	Address	Institute	Term	Resit	...
98052	Kunz	Jakob	Bichwil	G	5	N	
98053	Sutter	Ruth	Malhausen	G	5	N	
98054	Weber	Karl	Burgau	G	5	N	
98055	Beck	Karin	Lefendorf	G	5	N	

➲ http://www.gitta.info/IntroToDBS/en/html/DBApproaChar_trennDataApp.html

DB in a nutshell: 5. Data integrity

- The accuracy and consistency of the data stored are assured over the DB life cycle.
- DBMS is able to maintain data correctness.
- Integrity constraints
  - Entity integrity: “The price of any menu item must be greater than 0”
  - Referential integrity

artist_id	artist_name
1	Bono
2	Cher
3	Nuno Bettencourt

artist_id	album_id	album_name
3	1	Schizophonic
4	2	Eat the rich
3	3	Crave (single)

➲ More detail later in semester
➲ https://en.wikipedia.org/wiki/Referential_integrity

DB in a nutshell: 6. Transactions

- **Transactions:** atomic unit of interaction between a User and a DB that consists of one or more operations on the data:
  - Insertion of data into the DB
  - Modification of data in the DB
  - Deletion of data from the DB
  - Retrieval of data in the DB
- Example: two concurrent transactions
  - T_1: \$300 credit,
  - T_2: \$400 debit.

T_1	T_2	B	X	Y
		\$1000		
$X \leftarrow B$		\$1000	\$1000	
$X \leftarrow X + \$300$		\$1000	\$1300	
	$Y \leftarrow B$	\$1000	\$1300	\$1000
	$Y \leftarrow Y - \$400$	\$1000	\$1300	\$600
$B \leftarrow X$		\$1300	\$1300	\$600
	$B \leftarrow Y$	\$600		

Wrong, it should be \$900

➲ Worboys and Duckham (2004) GIS: A Computing Perspective, Second Edition, CRC Press, p. 42

DB in a nutshell: 6. Transactions

- Possible problems: updates lost, when concurrent atomic transactions are interleaved.
  - Same data invovled in multiple transactions
  - Example below: T_1 leaves no effects, as if it is overwritten.
- DBMS should stop such update errors
  - Transcation atomicity (all or nothing) & transaction independence
  - **commit**: signals a permanent change of the data when all constituent operations successfully completed
  - **rollback**: recovers the original state of the DB immediately prior to the transaction if problems occured

T_1	T_2	B	X	Y
		\$1000		
$X \leftarrow B$		\$1000	\$1000	
$X \leftarrow X + \$300$		\$1000	\$1300	
	$Y \leftarrow B$	\$1000	\$1300	\$1000
	$Y \leftarrow Y - \$400$	\$1000	\$1300	\$600
$B \leftarrow X$		\$1300	\$1300	\$600
	$B \leftarrow Y$	\$600		

Wrong, it should be \$900

➲ Worboys and Duckham (2004) GIS: A Computing Perspective, Second Edition, CRC Press, p. 42

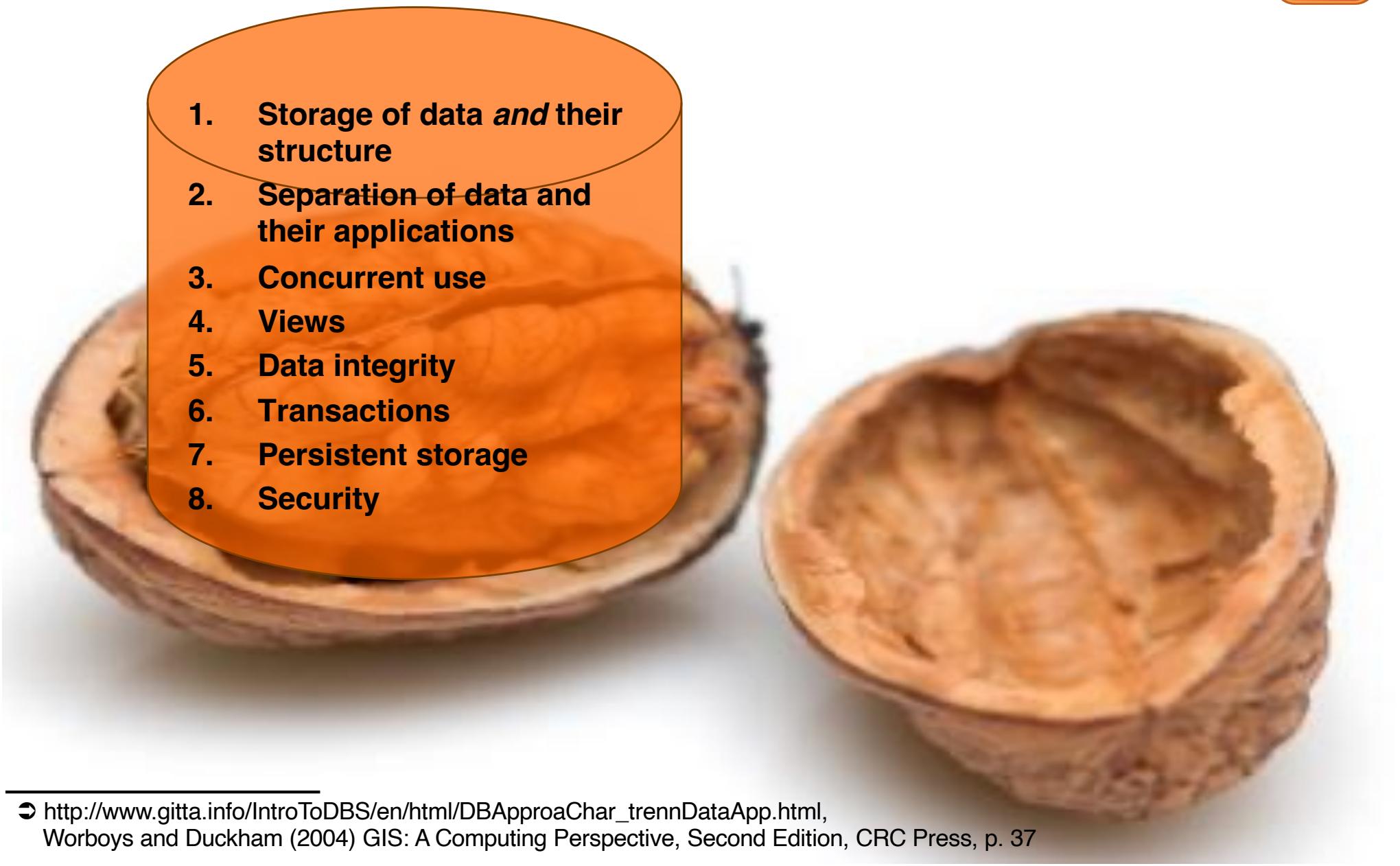
DB in a nutshell: 7. Persistent storage

- Data persistence means that in a DBMS all data is maintained as long as it is not deleted explicitly.
- The life span of data needs to be determined directly or indirectly by the user and must not be dependent on system features.
- Additionally data once stored in a database must not be lost.
- Recovery and backup

DB in a nutshell: 8. Security

- DB users
  - DB administrator (DBA)
  - DB designers
  - End users: users whose jobs require access to the DB for querying, updating, and generating reports.
  - system analysts and application programmers
- DBMS must be able to prevent data being used in unauthorized ways.
- Access operations: e.g., retrieval, update

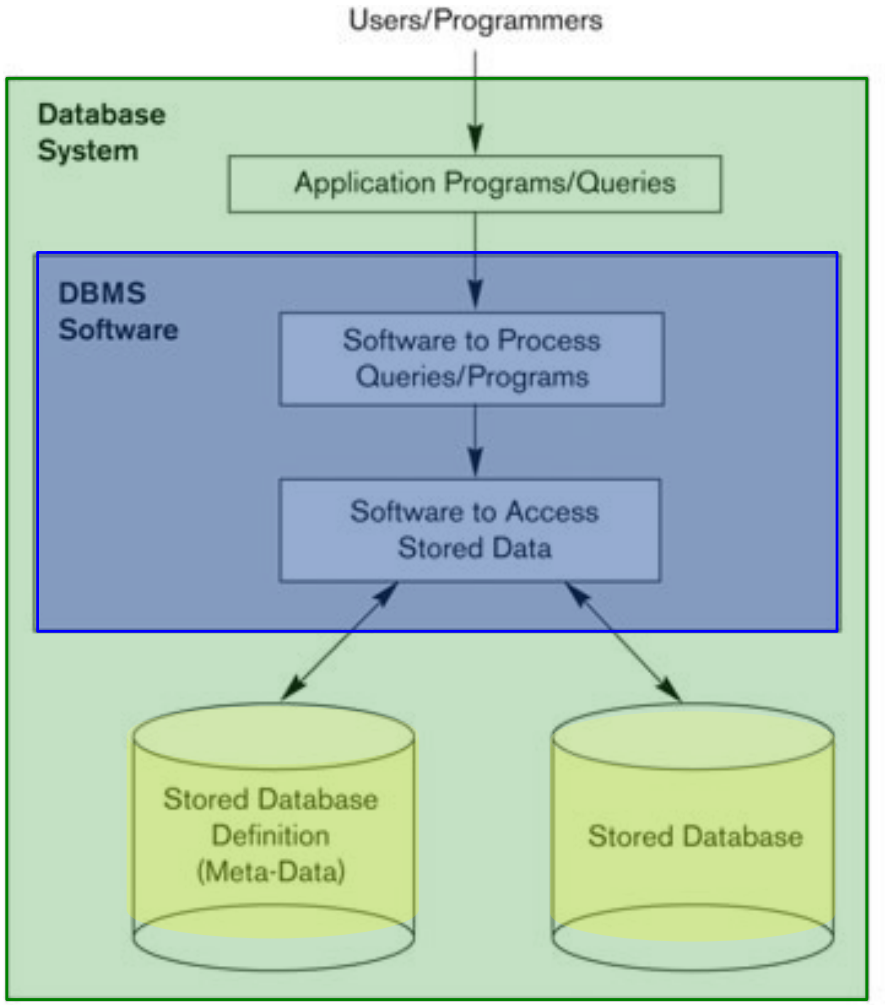
DB in a nutshell



➲ http://www.gitta.info/IntroToDBS/en/html/DBApproaChar_trennDataApp.html,
Worboys and Duckham (2004) GIS: A Computing Perspective, Second Edition, CRC Press, p. 37

Definitions

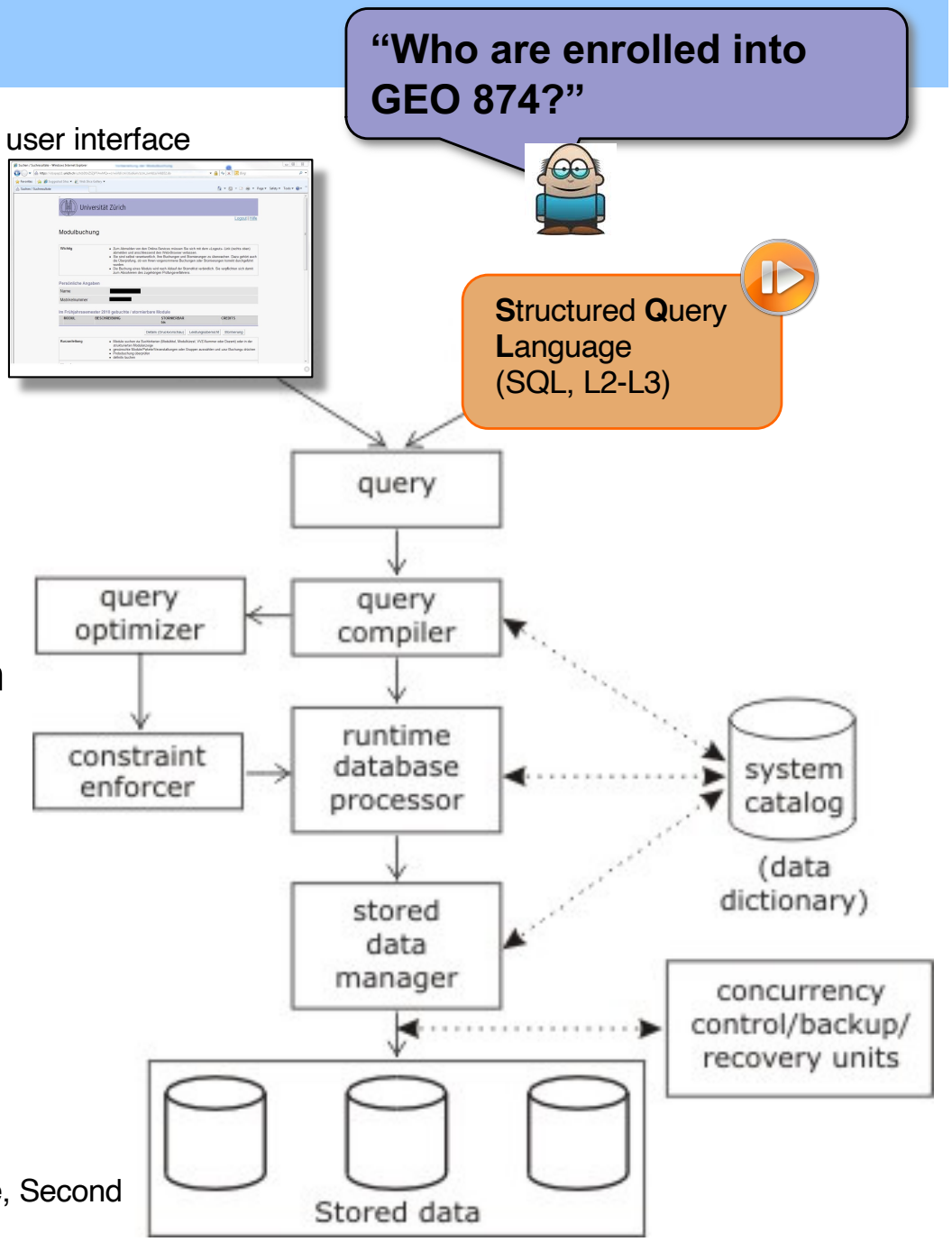
- **Database System** (DBS) = DB + DBMS
- **Database Management System** (DBMS) – (e.g., PostgreSQL)
  - Software tool to query, access and maintain a DB
- **Database** (DB)
  - Data – logically coherent collection of facts administered and accessed through the DBMS
  - Data Structure – the schemas
- Information system (e.g.,GIS)
  - Extends the DBS with tools for the querying, representation, transformation, visualization and analysis of data (e.g., spatial)



➲ Elmasri & Navathe (2014): Chapter 1.2, p.5.

Architecture of a DBMS

1. *Query compiler: parses and analyses the query, creates exec code*
2. *Query optimizer:* find efficient exec strategy
3. *Constraint enforcer:* enforces integrity constraints
4. *Runtime database processor:* executes code
5. *Stored data manager:* interacts with OS for physical access to data
6. *System catalog/data dictionary:* Metadata – info about the data model, the internal, conceptual & views
7. *Concurrency, control, backup, recovery units:* additional possible units



➲ Worboys and Duckham (2004) GIS: A Computing Perspective, Second Edition, CRC Press, p. 40

Introduction to Databases

1. Example: „Nutty Nuggets“
2. The Database Way...
   - „DB in a nutshell“
   - Database Management System (DBMS)
3. **Relational Databases**
   - Relational Model
   - Operations on relations
4. Pros- and Cons- of Databases
5. Products, Big players, Applications

3. Relational Databases

- There are a number of DB models:
  - **Relational DB**
  - Object-oriented DB (& also object-relational DBs)
  - NoSQL DBs: key-value, graph-based, Column, Document stores,... (http://en.wikipedia.org/wiki/NoSQL)

- Relational Model
  - Proposed by Ted Codd from IBM, 1970,
  - First implementation „IBM System R“ (1974), then Oracle
  - The **relational model** represents the database as a collection of mathematical *relations* (relational algebra), i.e., two-dimensional tables

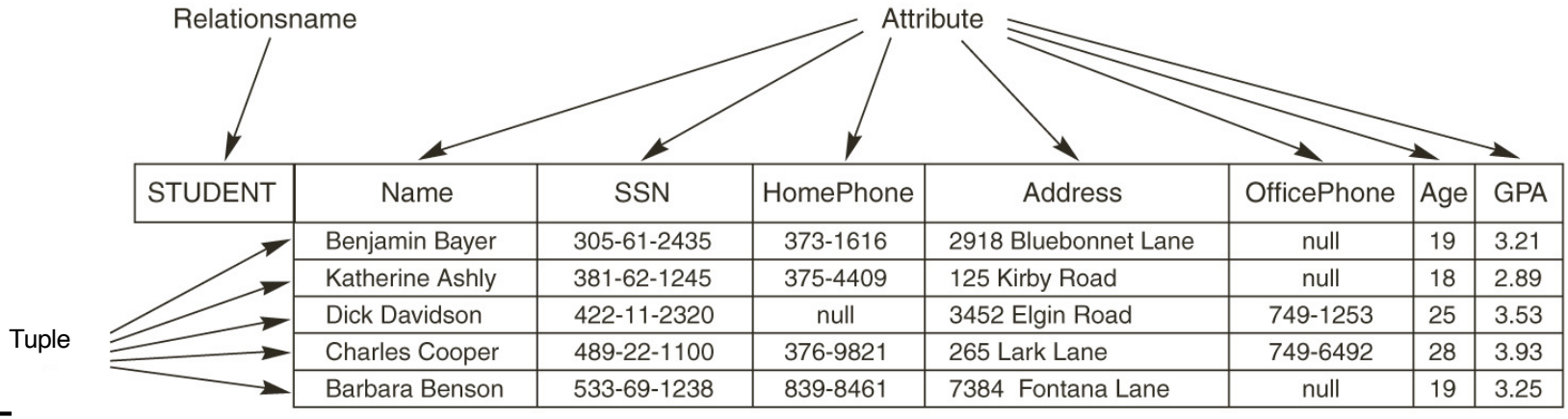
STUDENT	Name	StudentNumber	Class	Major
	Smith	17	1	CS
	Brown	8	2	CS

COURSE	CourseName	CourseNumber	CreditHours	Department
	Intro to Computer Science	CS1310	4	CS
	Data Structures	CS3320	4	CS
	Discrete Mathematics	MATH2410	3	MATH
	Database	CS3380	3	CS

➲ Codd, E.F. (1970). A Relational Model of Data for Large Shared Data Banks. Communications of the ACM 13 (6): 377–387. doi:10.1145/362384.362685

The Relational DB Model

- A relational DB is a collection of tables (= **relations**)
- Each relation/table has a name, and contains a number of **attributes (=fields, or columns)**
- Each attribute *A* has a **domain** (= **data type**), which defines what kinds of data the attribute can contain (e.g., integer, string, ...)
- A **tuple** (= **record**, or **row**) is a set of values from the domains of attributes $A_1..A_n$ from a given relation.
- Each tuple consists of values for each attribute.
- Each attribute in a tuple has a single value.



STUDENT	Name	SSN	HomePhone	Address	OfficePhone	Age	GPA
	Benjamin Bayer	305-61-2435	373-1616	2918 Bluebonnet Lane	null	19	3.21
	Katherine Ashly	381-62-1245	375-4409	125 Kirby Road	null	18	2.89
	Dick Davidson	422-11-2320	null	3452 Elgin Road	749-1253	25	3.53
	Charles Cooper	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
	Barbara Benson	533-69-1238	839-8461	7384 Fontana Lane	null	19	3.25

➲ Elmasri & Navathe (2014): chapter 3.1.1, p. 59.

Operations on Relations (Discrete Math: Relational algebra)

- 5 basic operations…
  - union
  - difference
  - product

  Set-Operations
  - project
  - select
- …3 derived operations (can be composed from above)
  - intersect
  - divide
  - join
- Structure and rules about operator use = **relational algebra**

(a)

STUDENT	FN	LN
	Susan	Yao
	Ramesh	Shah
	Johnny	Kohler
	Barbara	Jones
	Amy	Ford
	Jimmy	Wang
	Ernest	Gilbert

INSTRUCTOR	FNAME	LNAME
	John	Smith
	Ricardo	Browne
	Susan	Yao
	Francis	Johnson
	Ramesh	Shah

(b)

FN	LN
Susan	Yao
Ramesh	Shah
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert
John	Smith
Ricardo	Browne
Francis	Johnson

UNION (S or I or both)
STUDENT ∪ INSTRUCTOR

(c)

FN	LN
Susan	Yao
Ramesh	Shah

(d)

FN	LN
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilber

(e)

FNAME	LNAME
John	Smith
Ricardo	Browne
Francis	Johnson

➲ https://en.wikipedia.org/wiki/Relational_algebra

➲ Images: Elmasri & Navathe (2014)

Operations on Relations (Discrete Math: Relational algebra)

- 5 basic operations…
  - union
  - difference
  - product

  Set-Operations
  - project
  - select
- …3 derived operations (can be composed from above)
  - intersect
  - divide
  - join
- Structure and rules about operator use = **relational algebra**

(a)

STUDENT	FN	LN
	Susan	Yao
	Ramesh	Shah
	Johnny	Kohler
	Barbara	Jones
	Amy	Ford
	Jimmy	Wang
	Ernest	Gilbert

INSTRUCTOR	FNAME	LNAME
	John	Smith
	Ricardo	Browne
	Susan	Yao
	Francis	Johnson
	Ramesh	Shah

(b)

FN	LN
Susan	Yao
Ramesh	Shah
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert
John	Smith
Ricardo	Browne
Francis	Johnson

(c)

FN	LN
Susan	Yao
Ramesh	Shah

(d)

FN	LN
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilber

DIFFERENCE (S not I)
STUDENT - INSTRUCTOR

(e)

FNAME	LNAME
John	Smith
Ricardo	Browne
Francis	Johnson

DIFFERENCE (I not S)
INSTRUCTOR - STUDENT

➲ https://en.wikipedia.org/wiki/Relational_algebra

➲ Images: Elmasri & Navathe (2014)

Operations on Relations (Discrete Math: Relational algebra)

- 5 basic operations…
  - union
  - difference
  - product

  Set-Operations
  - project
  - select
- …3 derived operations (can be composed from above)
  - intersect
  - divide
  - join
- Structure and rules about operator use = **relational algebra**

FirstName	FN
	Esra
	Rolf

LastName	LN
	Suel
	Meile

FN	LN
Esra	Suel
Esra	Meile
Rolf	Suel
Rolf	Meile

Cartesian product of FirstName and LastName

➲ https://en.wikipedia.org/wiki/Relational_algebra

➲ Images: Elmasri & Navathe (2014)

Operations on Relations (Discrete Math: Relational algebra)

- 5 basic operations…
  - union
  - difference
  - product

  Set-Operations
  - project
  - select
- …3 derived operations (can be composed from above)
  - intersect
  - divide
  - join
- Structure and rules about operator use = **relational algebra**

(a)

STUDENT	FN	LN
	Susan	Yao
	Ramesh	Shah
	Johnny	Kohler
	Barbara	Jones
	Amy	Ford
	Jimmy	Wang
	Ernest	Gilbert

INSTRUCTOR	FNAME	LNAME
	John	Smith
	Ricardo	Browne
	Susan	Yao
	Francis	Johnson
	Ramesh	Shah

LN
Yao
Shah
Kohler
Jones
Ford
Wang
Gilbert

Project returns values of a subset of attributes/columns

➲ https://en.wikipedia.org/wiki/Relational_algebra

➲ Images: Elmasri & Navathe (2014)

Operations on Relations (Discrete Math: Relational algebra)

- 5 basic operations…
  - union
  - difference
  - product

  Set-Operations
  - project (subset of columns)
  - select (subset of rows)
- …3 derived operations (can be composed from above)
  - intersect
  - divide
  - join
- Structure and rules about operator use = **relational algebra**

(a)

STUDENT	FN	LN
	Susan	Yao
	Ramesh	Shah
	Johnny	Kohler
	Barbara	Jones
	Amy	Ford
	Jimmy	Wang
	Ernest	Gilbert

INSTRUCTOR	FNAME	LNAME
	John	Smith
	Ricardo	Browne
	Susan	Yao
	Francis	Johnson
	Ramesh	Shah

(b)

FN	LN
Susan	Yao
Ramesh	Shah
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilbert
John	Smith
Ricardo	Browne
Francis	Johnson

(c)

FN	LN
Susan	Yao
Ramesh	Shah

INTERSECTION (S and I)
STUDENT ∩ INSTRUCTOR

(d)

FN	LN
Johnny	Kohler
Barbara	Jones
Amy	Ford
Jimmy	Wang
Ernest	Gilber

(e)

FNAME	LNAME
John	Smith
Ricardo	Browne
Francis	Johnson

➲ https://en.wikipedia.org/wiki/Relational_algebra

➲ Images: Elmasri & Navathe (2014)

Operations on Relations (Discrete Math: Relational algebra)

- 5 basic operations…
  - union
  - difference
  - product

  Set-Operations
  - project
  - select
- …3 derived operations (can be composed from above)
  - intersect
  - divide
  - join
- Structure and rules about operator use = **relational algebra**

Divide: will return those tuples from relation A which are associated to **every** B's tuple, excluding the attributes in B

CourseReg	FN	CourseID
	Esra	874
	Esra	875
	Rolf	874

Course	CourseID
	874
	875

Who take all available courses?

CourseReg DIVIDE Course

FN
Esra

Division is NOT implemented in SQL

➲ https://en.wikipedia.org/wiki/Relational_algebra

➲ Images: Elmasri & Navathe (2014)

Operations on Relations (Discrete Math: Relational algebra)

- 5 basic operations…
  - union
  - difference
  - product

  Set-Operations
  - project
  - select
- …3 derived operations (can be composed from above)
  - intersect
  - divide
  - join
- Structure and rules about operator use = **relational algebra**

Person_Nr	LastName	FirstName	PLZ
1	Von Gunten	Reto	3000
2	Stofer	Christian	6330
3	Stofer	Silvia	6330
4	Ginzler	Christian	9000
5	Burghardt	Dirk	8001

PLZ	City
6330	Cham
9000	St. Gallen
8002	Zürich
6300	Zug
8400	Winterthur
8003	Zürich
3000	Bern
8001	Zürich
8051	Zürich

Person_Nr	LastName	FirstName	PLZ	PLZ	City
1	Von Gunten	Reto	3000	3000	Bern
3	Stofer	Silvia	6330	6330	Cham
2	Stofer	Christian	6330	6330	Cham
5	Burghardt	Dirk	8001	8001	Zürich
4	Ginzler	Christian	9000	9000	St. Gallen

➲ https://en.wikipedia.org/wiki/Relational_algebra

➲ Images: Elmasri & Navathe (2014)

Introduction to Databases

1. Example: „Nutty Nuggets“
2. The Database Way...
   - „DB in a nutshell“
   - Database Management System (DBMS)
3. Relational Databases
   - Relational Model
   - Operations on relations
4. **Pros- and Cons- of Databases**
5. Products, Big players, Applications

Pros- of a DBS

+ Controlling redundancy: Don‘t need to store the same data multiple times for different user groups.
+ Restricting unauthorized access: DB administrator (DBA), DB designers, end users, system analysts and application programmers
+ Concurrent use (multiple use), transactions and views
+ Data persistence, recovery and backup (Data loss management)
+ Providing storage structures and techniques for efficient query processing: indexes, buffering, caching, SQL, query processing and optimization, …
  + Deals with increasing volume of data
  + More and more complex queries
+ Representing complex relationships among data
+ Enforcing integrity constraints to ensure correctness of the data
+ Permitting inferencing and actions using rules: e.g., triggers
+ Flexibility, separation of data from their use and storage environments
  + Hardware or operation system updates
  + Faster application development

Cons- of a DBS

- High initial investment and setup costs (personnel, time, architecture...)
- Complexity: DBS operation is demanding
- Personnel: administrator needed
- Additional costs:
  - Hardware
  - Software
  - Administration

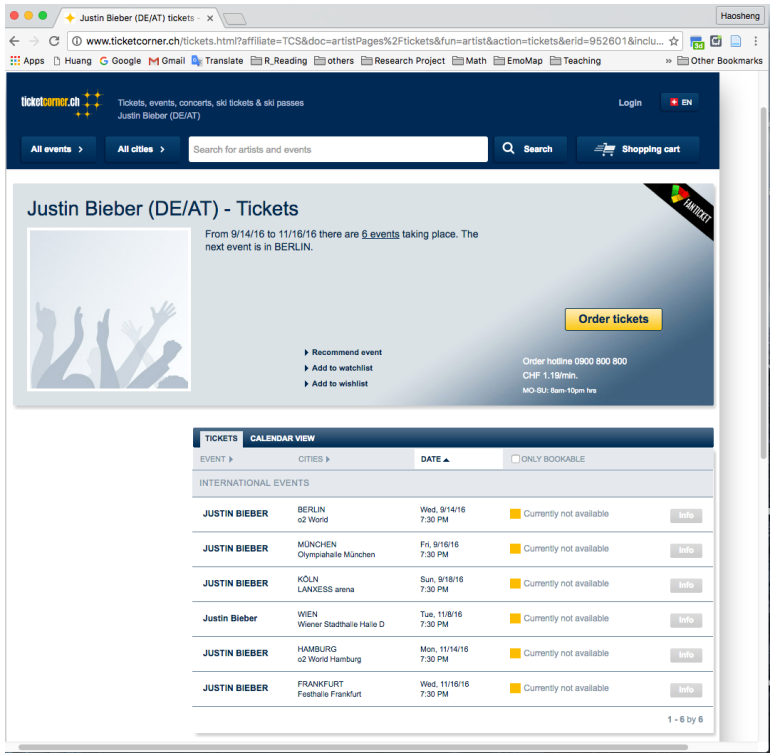
Cons- of a DBS

- High initial investment and setup costs (personnel, time, architecture...)
- Complexity: DBS operation is demanding
- Personnel: administrator needed
- Additional costs:
  - Hardware
  - Software
  - Administration

- When not to use a DBS
  - Simple, well-defined applications whose data are not expected to change at all
  - Embedded systems with limited storage and computing capacity
  - No multiple-user access to data

Exercise 1.2

- **Online-Ticket shop *revisited***
  - So– what are the Pros- and Cons- of storing data in databases Now?
  - Apply the concepts learnt, check your notes!



Introduction to Databases

1. Example: „Nutty Nuggets“
2. The Database Way...
   - „DB in a nutshell“
   - Database Management System (DBMS)
3. Relational Databases
   - Relational Model
   - Operations on relations
4. Pros- and Cons- of Databases
5. **Products, Big players, Applications**

Commercial and Open-Source Systems

DB "popularity"

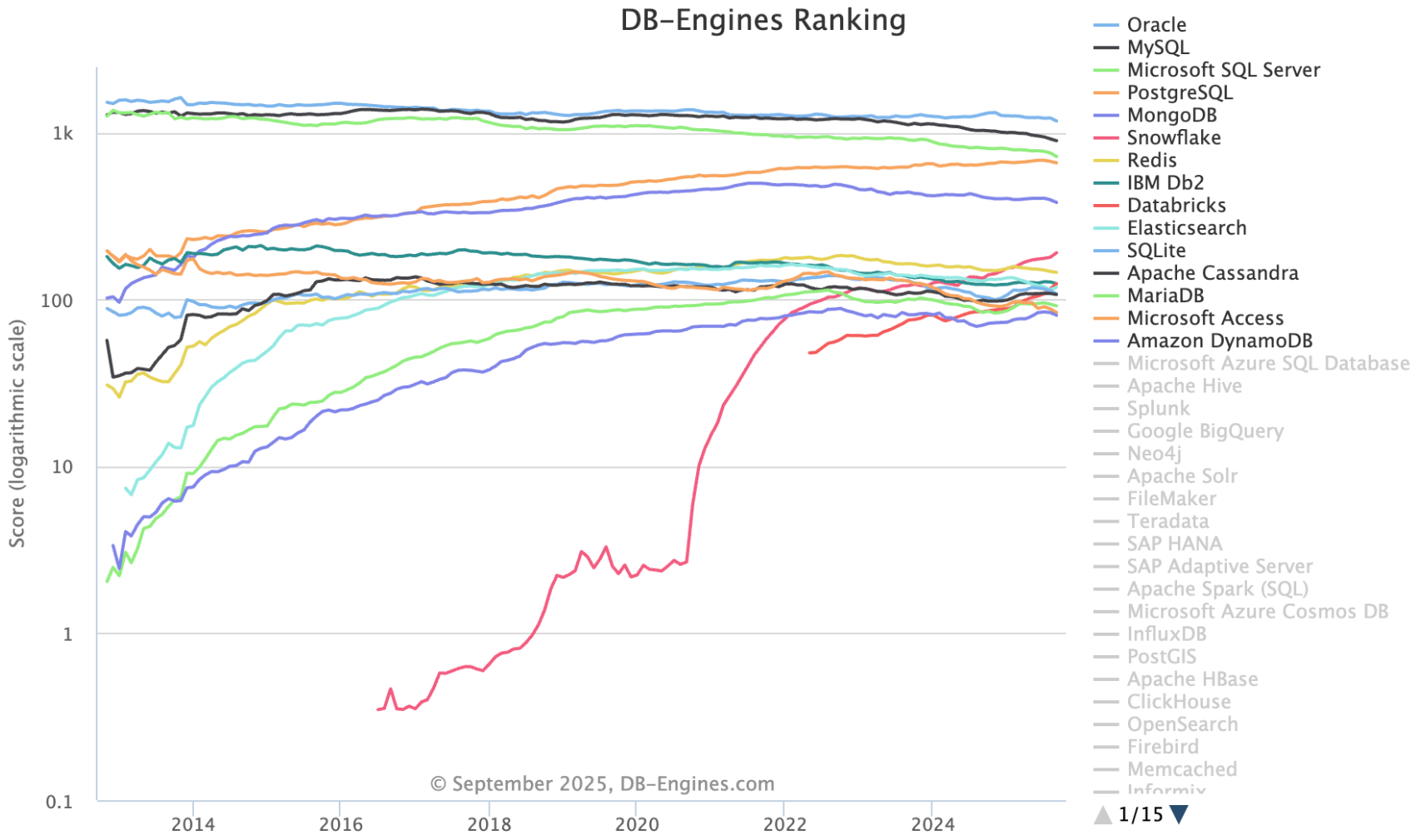
424 systems in ranking, September 2025

	Rank			DBMS	Database Model	Score		
	Sep 2025	Aug 2025	Sep 2024			Sep 2025	Aug 2025	Sep 2024
	1.	1.	1.	Oracle	Relational, Multi-model	1170.62	-50.08	-115.97
OS	2.	2.	2.	MySQL	Relational, Multi-model	891.77	-23.69	-137.72
	3.	3.	3.	Microsoft SQL Server	Relational, Multi-model	717.32	-36.84	-90.45
OS	4.	4.	4.	PostgreSQL	Relational, Multi-model	657.17	-14.08	+12.81
OS	5.	5.	5.	MongoDB	Document, Multi-model	380.50	-15.08	-29.74
	6.	6.	↑ 7.	Snowflake	Relational	190.19	+11.29	+56.47
OS	7.	7.	↓ 6.	Redis	Key-value, Multi-model	145.17	-2.02	-4.25
	8.	8.	↑ 9.	IBM Db2	Relational, Multi-model	124.19	-3.12	+1.14
	9.	9.	↑ 14.	Databricks	Multi-model	124.06	+8.25	+39.82
OS	10.	10.	↓ 8.	Elasticsearch	Multi-model	118.26	+3.99	-10.53
OS	11.	11.	↓ 10.	SQLite	Relational	107.88	-4.72	+4.53
OS	12.	12.	↓ 11.	Apache Cassandra	Wide column, Multi-model	106.98	-1.53	+8.04
OS	13.	13.	↑ 15.	MariaDB	Relational, Multi-model	91.46	-2.13	+8.02
	14.	14.	↓ 12.	Microsoft Access	Relational	83.61	-4.15	-10.15
	15.	15.	↑ 17.	Amazon DynamoDB	Multi-model	80.28	-3.20	+10.22
	16.	16.	16.	Microsoft Azure SQL Database	Relational, Multi-model	79.18	+3.34	+6.23
OS	17.	17.	↑ 18.	Apache Hive	Relational	76.10	+5.06	+23.02
	18.	18.	↓ 13.	Splunk	Search engine	75.77	+6.00	-17.26
	19.	19.	19.	Google BigQuery	Relational	66.00	+0.82	+13.33
OS	20.	20.	↑ 21.	Neo4j	Graph	53.78	-0.69	+11.10
OS	21.	21.	↑ 24.	Apache Solr	Search engine, Multi-model	36.76	+1.96	+2.61
	22.	↑ 23.	↓ 20.	FileMaker	Relational	34.44	+1.12	-10.76

➲ http://db-engines.com/en/ranking

Commercial and Open-Source Systems

DB “popularity” trend



➲ http://db-engines.com/en/ranking

Example applications of DBS

- Banking
  - Client management, account management, payment transactions, interest calculations, taxation
- Human resources databases (schools, companies)
  - Enrolment, course planning, room allocation, marking and credits, pay/leave/...
- Electronic Commerce, online shopping
  - Amazon, eBay, ..
- Industry
  - Stock management, bookings, task management
- Research (incl. Geographic information systems)
  - Measurement activities
  - Storage of spatial data: geometric, topologic and thematic facts

Summary

- Database management systems (DBMS) are tools for the administration and management of large quantities of data.
  - They separate data and applications, provide redundancy control and backup utilities, and allow for diverse applications.
  - The application of DBMS is demanding and complex, entails higher setup costs.
- The relational model is the foundation of the most common database architecture – the relational database.
- A relational DB is a collection of relations (SQL: Tables), each with tuples of values (SQL: records) with defined attributes (SQL: columns)
- Spatial databases are an important application area of DB technology

Glossary

Database (DB)	The actual stored data collection. A logically coherent collection of facts administered and accessed through the DMBS
Database management System (DBMS)	Management software. A software application that allows a user of an application to define, setup, access, manage and administer data access.
Database System (DBS)	DB + DBMS.
Transaction	A logical unit of data manipulation consisting of one or more operations on data.
Relation	A collection of lists of logically consistent values (tuples, in SQL: rows).
Relational model	Represents the database as a collection of mathematical relations (“tables of values”).

Later, in lab

- 10:15-12:00

- Y25-J-09, Y25-J-10

- **PostgreSQL**

- Working environment & graphical user interface (GUI)
- Examples of database, tables, …

Next Friday, in GEO 874: L2