Data Management
01 Introduction and Overview

Matthias Boehm

Graz University of Technology, Austria
Computer Science and Biomedical Engineering
Institute of Interactive Systems and Data Science
BMVIT endowed chair for Data Management

Last update: Oct 07, 2019

Announcements/Org

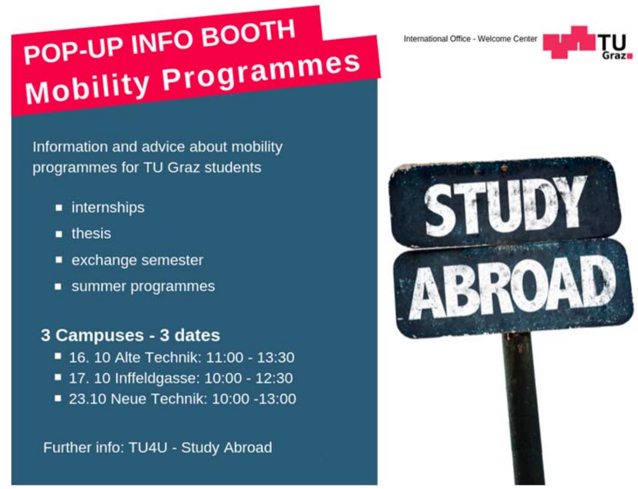
- **#1 Video Recording**
  - Link in TeachCenter & TUbe (lectures will be public)
- **#2 CS Talks x5** (Oct 15, 5pm, Aula Alte Technik)
  - Margarita Chli (ETH Zurich)
  - Title: How Robots See – Current Challenges and Developments in Vision-based Robotic Perception

- **#3 Course Intro International Entrepreneurship**
  - Understanding of intern. business, markets, and people
  - Lecturer: Univ.-Prof. Dr. techn. Hongying Foscht
  - Beginning Oct 9, 2019; 4 ECTS, 706.319

- **#4 Master Thesis – JOANNEUM RESEARCH Health**
  - **Thesis topic:** Development and validation of a hybrid decision model to identify frailty in older adults with care needs in geriatric care facilities
  - 60% part-time employment JOANNEUM RESEARCH, 8 months, € 831

Announcements/Org, cont.

- **#5 Study Abroad Info**
  - **Oct 17, 10am** @ Inffeldgasse
  - Internships, master theses, study courses, summer programs



- **#6 Workshop - Focus on FAIR**
  - **Nov 7**, **9.30am – 4pm** (all day)
  - Student reach-out BS, MS, PhD
  - Invited speakers from EGI, TU Delft, Uni. Vienna, Uni. Barcelona, UCL, and TU Graz/Know-Center

Agenda

- **Data Management Group**
- **Course Motivation, Goals, and Outline**
- **Course Organization and Logistics**
- **History of Data Management**

Data Management Group

About Me

- **09/2018 TU Graz**, Austria
  - BMVIT endowed chair for data management
  - **Data management for data science**
    (ML systems internals, end-to-end data science lifecycle)

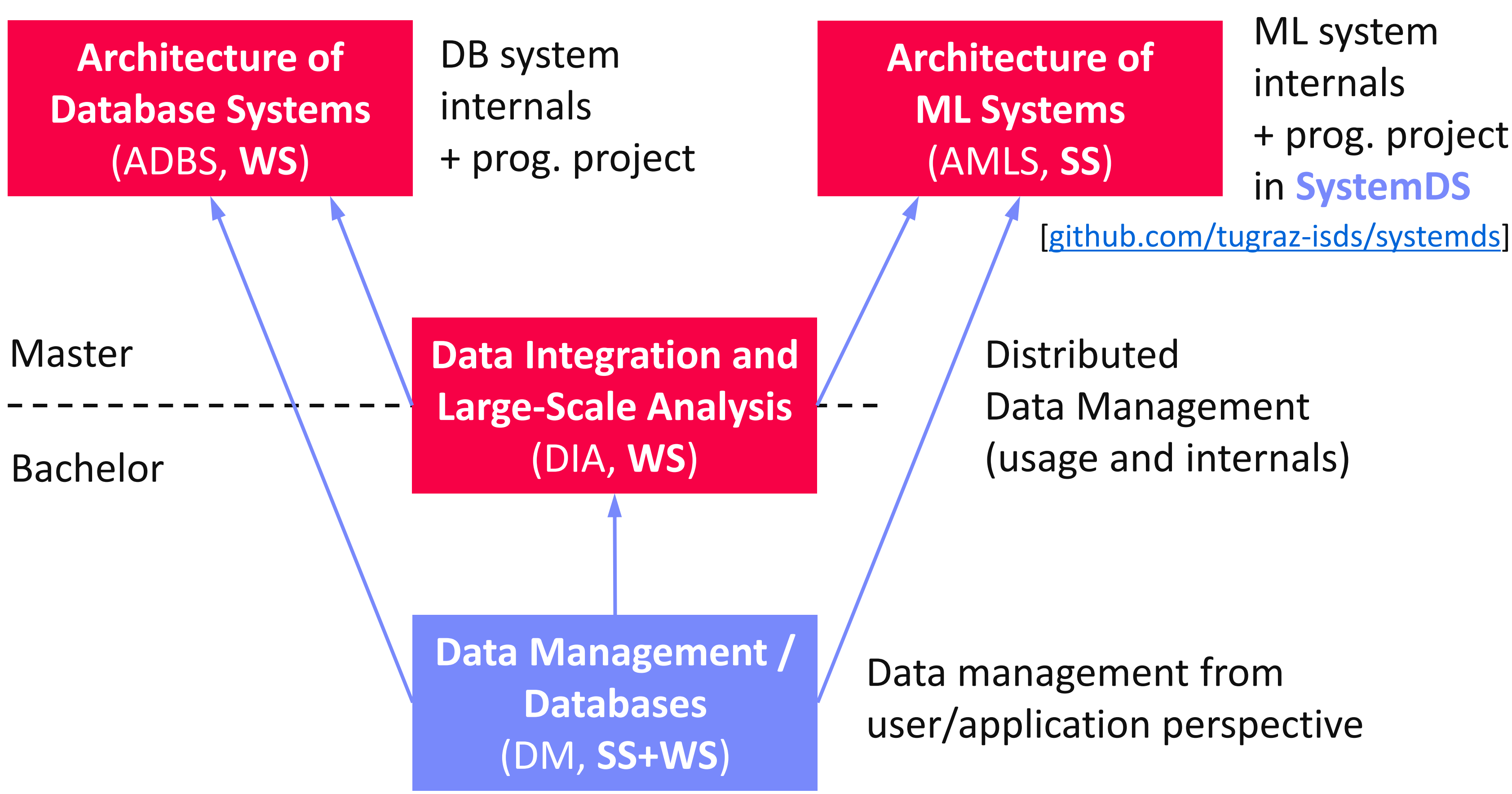
https://github.com/tugraz-isds/systemds

- **2012-2018 IBM Research – Almaden**, USA
  - Declarative large-scale machine learning
  - Optimizer and runtime of **Apache SystemML**

- **2011 PhD TU Dresden**, Germany
  - Cost-based optimization of integration flows
  - Systems support for time series forecasting
  - In-memory indexing and query processing

DB group

Data Management Courses

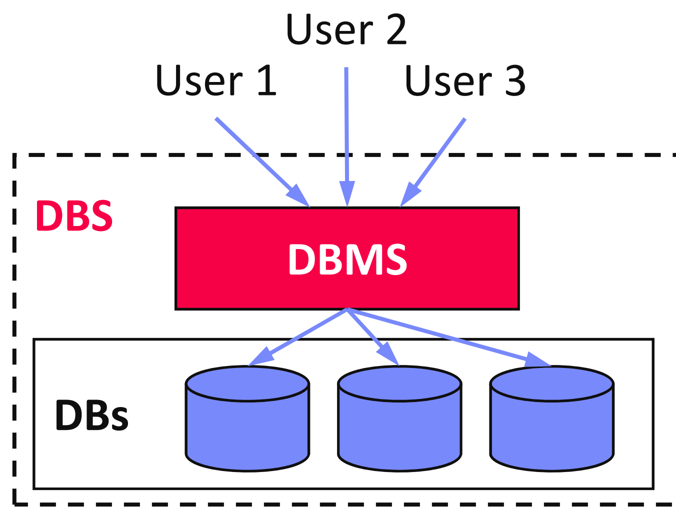


Course Motivation, Goals, and Outline

Database Systems and Modern Data Management

Definition and Impact

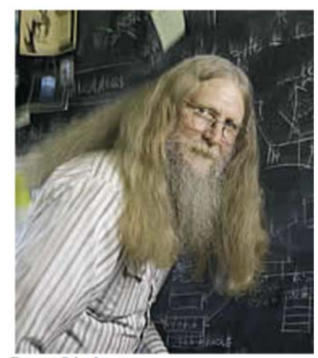
- **Def: Database System**
  - Overall system of DBMS + DBs
  - DBMS: Database Management System (SW to handle DBs)
  - DBs: Database (data/metadata collection of conceptual mini-world)
  - Note: DB also a short for DBS/DBMS



- **Importance in Practice**
  - Market Volume: **10-100B $US**
  - Foundation of many applications in various domains

"Relational databases are the foundation of western civilization"

[M. Winslett: Bruce Lindsay speaks out: […]. SIGMOD Record 34(2), **2005**]



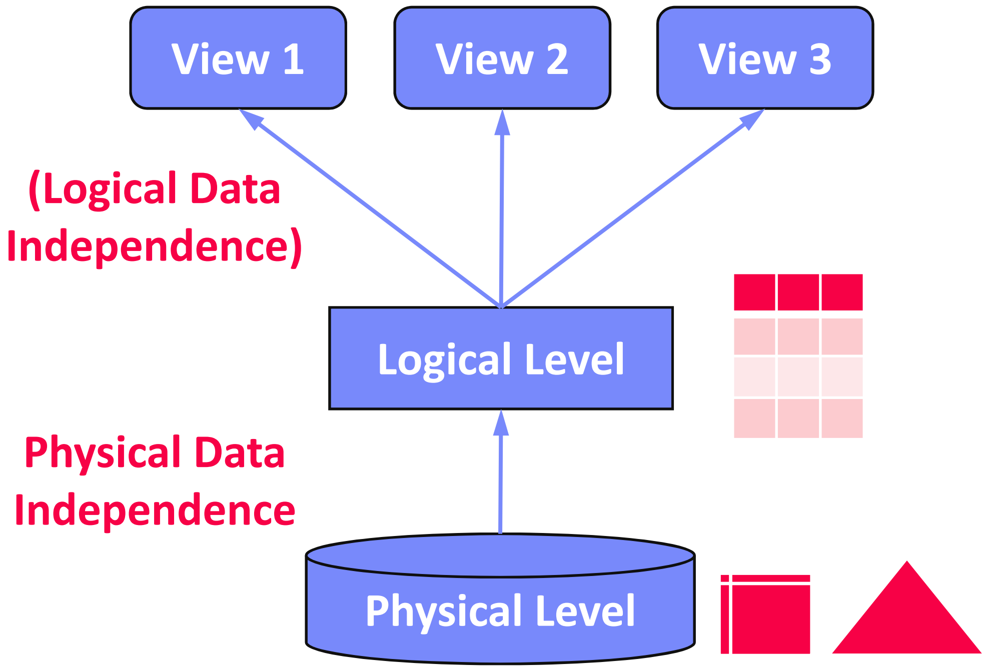
Bruce Lindsay

Motivation Database Systems

- **Application development and maintenance costs**
  - **Declarative queries** (what not how) and **data independence**
  - Efficient, correct, and independent data organization, size, access
- **Multi-user operations and access control**
  - Synchronization of **concurrent user queries and updates**
  - Enforce access control (e.g., permissions on tables, views)
- **Consistency and data integrity**
  - Eliminates **redundancy** and thus, enforces consistency
  - Enforces **integrity constraints** (e.g., semantic rules)
- **Logging and Recovery**
  - Recovery of consistent state after HW or SW failure
- **Performance and Scalability**
  - **High performance** for large datasets or high transaction throughput
  - Scale to large datasets with low memory requirements

Data Independence

- **Three Layer ANSI-SPARC Architecture**
  - External schemas (external level)
  - Conceptual schema (logical level)
  - Internal schema (physical level)



- **Types of Data Independence**
  - Logical data independence (external views and applications independent of logical data model)
  - Physical data independence (logical data model independent of underlying data organization)

Goals

- **Course Goals**
  - **A:** Understanding of database systems **from user perspective**
  - **B:** Understanding of modern means of data management **from user perspective**

	INF.01017UF (VO) Data Mgmt.	**INF.02018UF (KU) Data Mgmt.**
706.010 (VU) Databases	Part A 9 Lectures	Part A 3 Exercises
3(2) ECTS	Part B 3 Lectures	Part B 1 Exercise
	3 ECTS	1 ECTS

- **Meta Goals**
  - Understand, use, debug, and evaluate data management systems
  - Awareness of system alternatives and their tradeoffs
  - Fundamental concepts as basis for advanced courses and other areas

Part A: Database System Fundamentals

- **01 Introduction and Overview** [Oct 07]
- **02 Conceptual Architecture and Design** [Oct 14]
- **03 Data Models and Normalization** [Oct 21]

→ Exercise 1: Data Modeling **[Nov 05]**

- **04 Relational Algebra and Tuple Calculus** [Oct 28]
- **05 Query Languages (SQL)** [Nov 04]
- **06 APIs (ODBC, JDBC, OR frameworks)** [Nov 11]

→ Exercise 2: Queries **[Nov 26]**

- **07 Physical Design and Tuning** [Nov 18]
- **08 Query Processing** [Nov 25]
- **09 Transaction Processing and Concurrency** [Dec 02]

→ Exercise 3: Tuning **[Dec 20]**

Part B: Modern Data Management

- **10 NoSQL (key-value, document, graph)** [Dec 09]
- **11 Distributed file systems and object storage** [Jan 13]
- **12 Data-parallel computation (MapReduce, Spark)** [Jan 13]
- **13 Data stream processing systems** [Jan 20]

→ Exercise 4: Spark **[Jan 21]**

- **14 Q&A and exam preparation** [Jan 27]
- **Final written exam** [Jan 30 5.30pm, Jan 31 5.30pm]

Course Organization

Basic Course Organization

- **Staff**
  - **Lecturer:** Univ.-Prof. Dr.-Ing. Matthias Boehm, ISDS
  - **Teaching Assistants:** Dardan Dermaku, Olga Ovcharenko, Oliver Nikolic, Melanie Willfurth

- **Language**
  - Lectures and slides: **English**
  - Communication and examination: **English**/**German**
- **Course Format**
  - DM VO + KU 2/1 (**3+1 ECTS**), DB VU 1/1 (**3(2) ECTS**)
  - **Weekly lectures** (**start 6.10pm**, including **Q&A**), **attendance optional**
  - **4/3 exercises** (introduced in lecture) as individual assignments
  - **Recommended papers** for additional reading on your own

Course Logistics

- **Communication**
  - **Informal language** (first name is fine)
  - Please, **immediate feedback** (unclear content, missing background)
  - **Newsgroup:** news://news.tugraz.at/tu-graz.lv.dbase (email for private issues)
  - Office hours: **Mo 3pm**, or after lecture

- **Website**
  - https://mboehm7.github.io/teaching/ws1920_dbs/index.htm
  - All course material (lecture slides, exercises) and dates

- **Exam**
  - **Completed mandatory exercises** (Nov 05, Nov 26, Dec 20, [Jan21])
  - **Final written exam** (Jan 30 5.30pm, Jan 31 5.30pm, HS i13)
  - **DB Grading** (30% exercises, 70% final)

Course Logistics, cont.

- **Exercises**
  - Written and programming assignments, submitted through **TeachCenter**
  - Assignments have **25 points + 5 bonus points**
  - Assignment **completed if >50% points in total** (**vs last semester**)
  - Deadlines are important (**at most 7 late days in total**)
  - Individual assignments (**academic honesty / no plagiarism**)

- **SW Tools and Languages**
  - **Open Source PostgreSQL** DBMS (setup on your own)
  - Distributed FS / object storage and **Apache Spark** for distributed computation
  - Languages for local/distributed programs: e.g., C, C++, **Java**, **Scala** or **Python**

Exercises: Airlines and Flights

New

- **Dataset**
  - Public-domain, derived (parsed, cleaned) from the **OpenFlights Dataset**
  - Clone or download your copy from https://github.com/tugraz-isds/datasets.git

- **Exercises**
  - **01** Data modeling (relational schema)
  - **02** Data ingestion and SQL query processing
  - **03** Tuning, query processing, and transaction processing
  - **04** Large-scale data analysis (distributed data ingestions and query processing)

Airlines.csv: The Airlines file contains the airlines informatior

```
#Name, IATA, ICAO, Country, Active
Austrian Airlines,OS,AUA,Austria,Y
Turkish Airlines,TH,THY,Turkey,Y
Lufthansa,MH,DLH,Germany,Y
```

Airports.csv: The Airports file contains the airports informati

```
#Name, City, Country, IATA, ICAO, Latitude, Logtitude,
Goroka Airport,Goroka,Papua New Guinea,GKA,AYGA,-6.0816
Kaduna Airport,Kaduna,Nigeria,KAD,DNKA,10.6960000991821
Brussels Airport,Brussels,Belgium,BRU,EBBR,50.90140151
```

Routes.csv: The Routes file contains the flights information. I

```
#Airline, Departure, Arrival, Plane
NF,NUS,VLI,YN2;DHT;BNI
Y9,IFN,MRX,TU3
6R,MJZ,YKS,TU3;AN4
3R,ASF,DME,SU9
```

Planes.csv: The Planes file contains the planes information. I

```
#Name, IATA, ICAO
Aerospatiale SN.601 Corvette,NDC,S601
Airbus A380-800,388,A388
Antonov AN-12,ANF,AN12
Boeing 737-400,734,B734
```

Literature

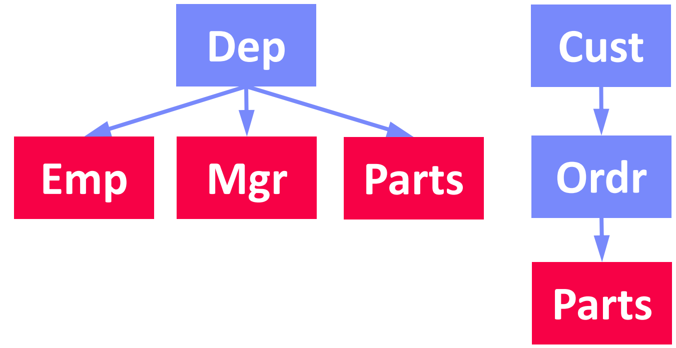
- **Not needed for lectures / exercises** (course is self-contained), **but second perspective on covered topics of first part**
- Raghu Ramakrishnan, Johannes Gehrke: Database Management Systems (3. ed.). McGraw-Hill 2003, ISBN 978-0-07-115110-8, pp. I-XXXII, 1-1065
- Jeffrey D. Ullman, Jennifer Widom: A first course in database systems (2. ed.). Prentice Hall 2002, ISBN 978-0-13-035300-9, pp. I-XVI, 1-511
- Ramez Elmasri, Shamkant B. Navathe: Fundamentals of Database Systems, 3rd Edition. Addison-Wesley-Longman 2000, ISBN 978-0-8053-1755-8, pp. I-XXVII, 1-955
- Alfons Kemper, André Eickler: Datenbanksysteme - Eine Einführung, 10. Auflage. De Gruyter Studium, de Gruyter Oldenbourg 2015, ISBN 978-3-11-044375-2, pp. 1-879

History of Data Management

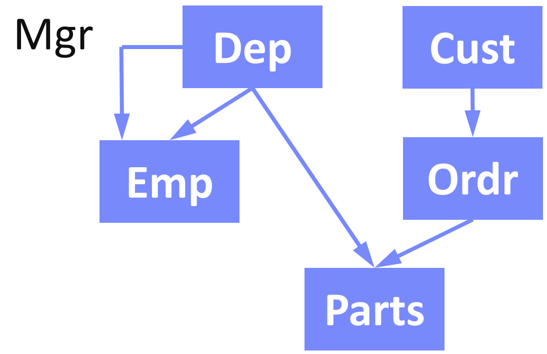
History 1960/70s (pre-relational)

CODASYL … Conference on Data Systems Languages

- **Hierarchical Model**
  - Tree of records
  - E.g., IBM Information Management System (IMS) – IMS 15 (Oct 2017)



- **Network Model**
  - CODASYL (COBOL, DB interfaces)
  - Graph of records
  - Charles Bachman (Turing Award '73)
  - E.g., Integrated Data Store (IDS)



- **Pros and Cons** (see NoSQL Doc-Stores)
  - Performance by **directly traversing static links**
  - **Duplicates** → inconsistencies on updates, **data dependence**

History 1970/80s (relational)

SQL Standard (SQL-**86**)

↑

SEQUEL

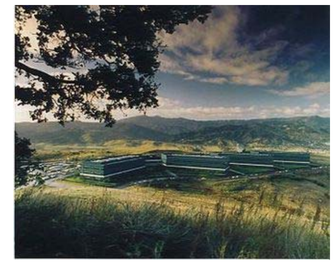
Oracle, IBM DB2, Informix, Sybase → MS SQL

QUEL - - - → **SEQUEL**

↑ QUEL | ↑ SEQUEL

Ingres @ UC Berkeley (Stonebraker et al., Turing Award '14)

System R @ IBM Research – Almaden (Jim Gray et al., Turing Award '98)



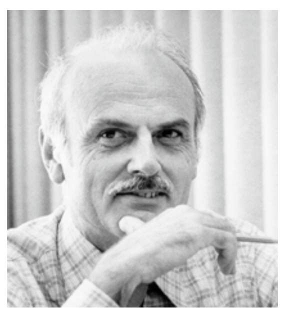
Tuple Calculus

Relational Algebra

Relational Model

Goal: **Data Independence** (physical data independence)

- Ordering Dependence
- Indexing Dependence
- Access Path Depend.



Edgar F. "Ted" Codd @ IBM Research (Turing Award '81)

[E. F. Codd: A Relational Model of Data for Large Shared Data Banks. Comm. ACM 13(6), **1970**]

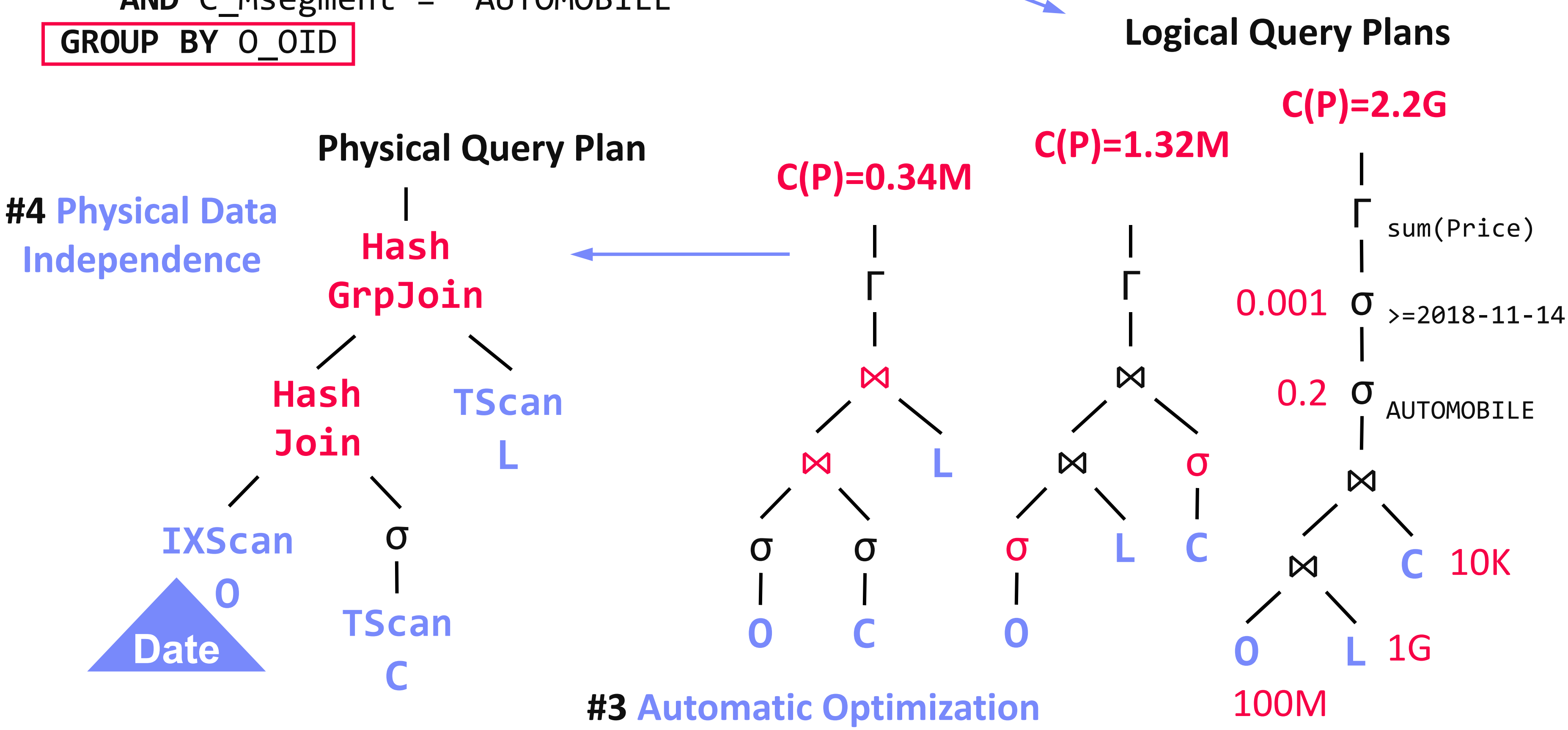
Success of SQL / Relational Model

Query:

```
SELECT O_OID, sum(O_Price)
FROM Orders, Lineitem, Customer
WHERE O_OID = L_OID AND O_CID = C_CID
   AND O_Odate >= '2018-11-14'
   AND C_Msegment = 'AUTOMOBILE'
GROUP BY O_OID
```

#1 Declarative: what not how

#2 Flexibility: closure property → composability



Excursus: PostgreSQL

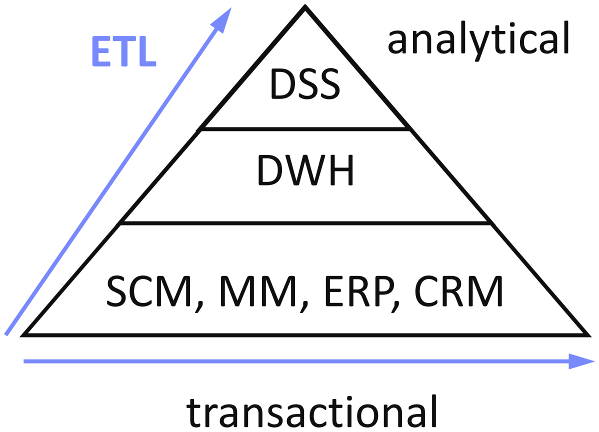
- **History of PostgreSQL** (used in the exercises)
  - Postgres is the successor project of commercialized Ingres
  - Focus on abstract data types, commercialized as Illustra
  - Prototype w/ SQL open sourced as **Postgres95** → **PostgreSQL**
  - Heavily used as **basis for research projects / startups**

- **Recommended Reading**
  - Michael Stonebraker: The land sharks are on the squawk box. Commun. ACM 59(2): 74-83 (2016), Turing Award Lecture, https://dl.acm.org/citation.cfm?doid=2886013.2869958
  - **Video:** http://www.youtube.com/watch?v=sEPTZVGk3WY
  - **Slides:** http://vldb.org/2015/wp-content/uploads/2015/09/stonebraker.pdf

History 1980/90/2000s

OLTP … Online Transaction Processing
OLAP … Online Analytical Processing
ETL … Extract, Transform, Load

- **Enterprise DBMS**
  - **Heavy investment in research and development** ➔ adoption
  - Oracle, IBM DB2, Informix, Sybase, MS SQL, PostgreSQL, MySQL
  - **Other technologies:** OODBMS, Multimedia, Spatiotemporal, Web, XML
- **Information/Data Warehousing (DWH)**
  - **Workload separation into OLTP and OLAP**
  - **Classical DWH architecture:** operational, staging, DWH, data marts + mining
  - **ETL Process** (Extract, Transform, Load)



- **Different Personas**
  - Domain Experts (e.g., BI Tools, SAP R/3)
  - DB Application Developers (e.g., ABAP)
  - DB Developers and DB Admins

→ SQL

History 2000s / Early 2010s

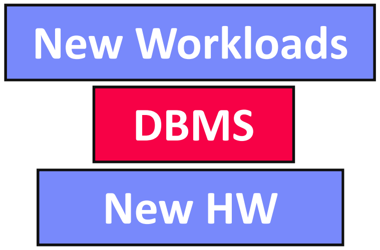
[M. Stonebraker, S. Madden, D. J. Abadi, S. Harizopoulos, N. Hachem, P. Helland: **The End of an Architectural Era** (It's Time for a Complete Rewrite). **VLDB 2007**]


- **Specialized Systems**
  - **Column stores + compression** for OLAP
  - **Main memory systems** for OLTP and OLAP
  - Data streaming, scientific and graph databases
  - Information extraction / retrieval, and XML
- **Other Research Trends**
  - Approximate QP / **Adaptive QP** / tuning tools
  - **Large-scale data management** (DFS, MR) / **cloud computing**
- **Toward Flexible, Large-Scale Data Management** (DWH … a bygone era)
  - **MAD Skills (magnetic, agile, deep)**, MADlib
  - Integration of R, Python in data analysis
  - Open data and its integration
  - Query processing over raw data files

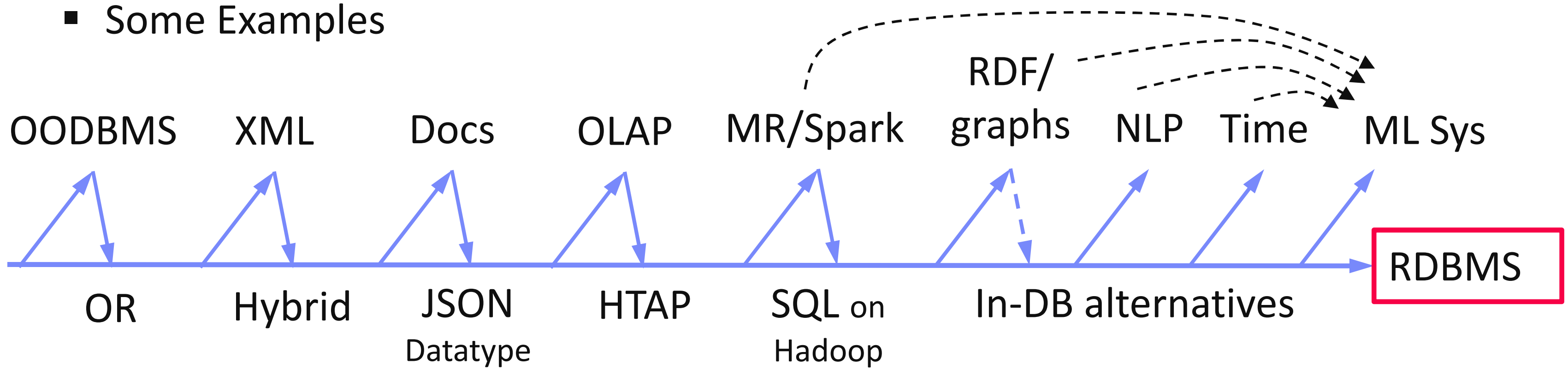
[J. Cohen, B. Dolan, M. Dunlap, J. M. Hellerstein, C. Welton: **MAD Skills**: New Analysis Practices for Big Data. **PVLDB 2(2) 2009**]


History 2010s – Present

- **Two Key Drivers of DB Research**
  - New analysis workloads (NLP, key/value, RDF/graphs, documents, time series, ML) and applications
  - New HW/infrastructure (multi-/many-core, cloud, scale-up/ scale-out, NUMA/HBM, RDMA, SSD/NVM, FPGA/GPU/ASIC)



- **Excursus: A retrospective view of specialized systems**
  - Goal **#1**: Avoid boundary crossing ➔ **General-purpose**
  - Goal **#2**: New workload + Performance ➔ **Specialized systems**
  - Some Examples



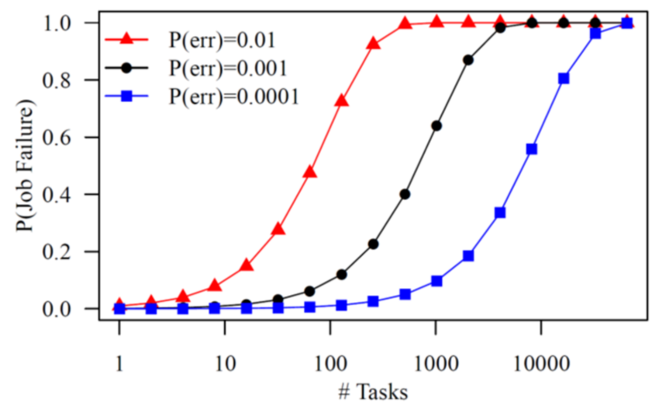
History 2010s – Present (2)

- **Motivation NoSQL Systems**
  - **Flexible schema** (no upfront costs), **scalability**, or specific data types
  - Relaxed **ACID** (atomicity, consistency, isolation, durability) requirements
    ➔ **BASE** (basically available, soft state, eventual consistency)

- **Example NoSQL Systems** (local and distributed)**:**
  - **Key/Value-Stores:** simple put/get/delete, massive scalability
  - **Document-Stores:** store nested documents (tree)
  - **RDF Stores:** store subject-predicate-object triples
  - **Graph DBs:** store nodes/edges/attributes, vertex-centric
  - **Time Series DBs:** store sequences of observations

History 2010s – Present (3)

- **Motivation Large-Scale Data Management**
  - Massive scalability (data/compute) on demand, fault tolerance, flexibility
  - Example Facebook 2014: **300PB** DWH, **600TB** daily ingest
  - Cost-effective commodity hardware
  - Error rate increases with increasing scale



- **Examples Large-Scale Data Management**
  - Distributed file systems w/ replication (e.g., GPFS, HDFS)
  - Cloud object storage (e.g., Amazon s3, OpenStack Swift)
  - Data-parallel data analysis with Spark/Flink, incl streaming
  - Automatic cloud resource elasticity (pay as you go)

Data Lake

Summary and Q&A

- **#1 Database Systems**
  - Mature and established technology → broadly applicable & eco system
  - **General concepts:** abstraction, data modeling, query optimization & processing, transaction processing and recovery, physical design and tuning
- **#2 Modern Data Management**
  - Multiple specialized systems for specific scale / data types
  - General trend toward less upfront cost, flexibility, and higher scalability

→ Variety of data management tools → Course meta goals

  - **Understand, use, debug**, and evaluate data management systems
  - **Fundamental concepts** as basis for advanced courses and other areas

- **Upcoming**
  - **02 Conceptual Architecture and Design** [Oct 14] (ER Diagrams)
  - **03 Data Models and Normalization** [Oct 21] (ERD -> Relational Model)