

UNIV2V96 Transfer Research Initiative

Data Science in Practice: Data Management

Karl Ho

School of Economic, Political and Policy Sciences

University of Texas at Dallas

Overview

- What is database?
- Brief history of database systems
- Architecture of database
- Data models
- File processing systems vs. DBMS
- Database types
- Relational database and NoSQL
- Query processing

Database

A database is a structured collection of data about entities and their relationships.

- Object - entities (e.g. individuals, organization, etc.)
- Relationship - (John works in WTO)

Database use

- Computer Science
- Business / Management
- Physical sciences
- Policy Sciences
- Social Sciences

History of Database Systems

1. 1950s and early 1960s
 1. Data processing using magnetic tapes for storage
 2. Tapes provided only sequential access
 3. Punched cards for input

History of Database Systems

1. Late 1960s and 1970s

1. Hard disks allowed direct access to data

2. Network and hierarchical data models in widespread use

- Edgar (Ted) Frank Codd defines the relational data model
- IBM Research begins System R prototype
- UC Berkeley (Michael Stonebraker) begins Ingres prototype
- Oracle releases first commercial relational database
- High-performance (for the era) transaction processing

History of Database Systems

1. 1980s

1. Research relational prototypes evolve into commercial systems
 1. SQL becomes industrial standard
2. Parallel and distributed database systems
 1. Wisconsin, IBM, Teradata
3. Object-oriented database systems

History of Database Systems

1. 1990s

1. Large decision support and data-mining applications
2. Large multi-terabyte data warehouses
3. Emergence of Web commerce

History of Database Systems

1. 2000s

1. Big data storage systems

1. Google BigTable, Yahoo PNuts, Amazon,
2. “NoSQL” systems.

2. Big data analysis: beyond SQL

1. Map reduce and friends

History of Database Systems

1. 2010s

1. SQL reloaded

1. SQL front end to Map Reduce systems
2. Massively parallel database systems
3. Multi-core main-memory databases

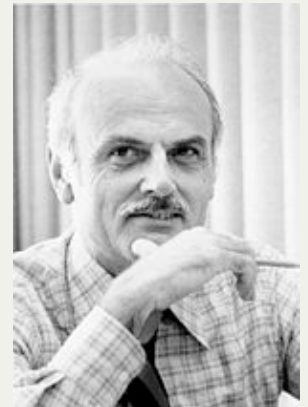
History of Database Systems

“Those that do not understand the mistakes of their ancestors will end up repeating them”

- Michael Stonebraker

Edgar Frank Codd (1923-2003)

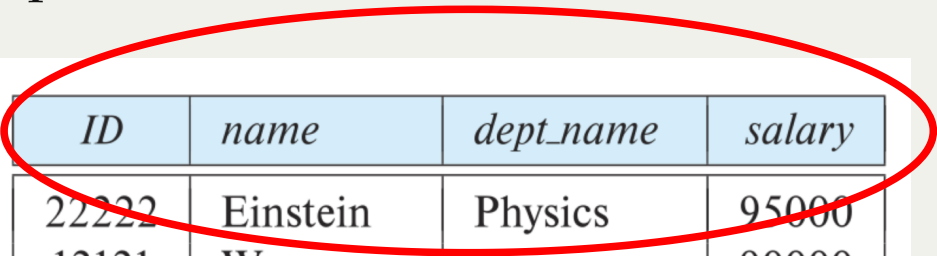
- Most known as Ted Codd
- Turing Award (1981)
- British born American computer scientist
- Education: Oxford, Michigan
- Work: IBM
- Invented "*Relational*" database when working in IBM in the early 1970s
- Famous paper:
"A Relational Model of Data for Large Shared Data Banks"



<https://history.computer.org/pioneers/codd.html>

Relational model

- All the data is stored in various tables.
- Example of tabular data in the relational model



<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

Relational model

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	60000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

(a) The *instructor* table

<i>dept_name</i>	<i>building</i>	<i>budget</i>
Comp. Sci.	Taylor	100000
Biology	Watson	90000
Elec. Eng.	Taylor	85000
Music	Packard	80000
Finance	Painter	120000
History	Painter	50000
Physics	Watson	70000

(b) The *department* table

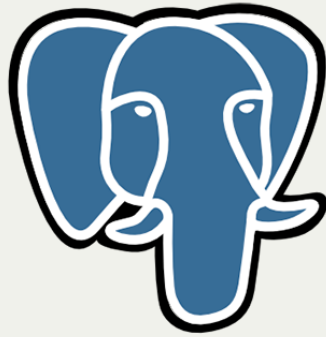
Non-nested tables

Michael Stonebraker

- Turing award laureate (2014)
- Education: Princeton, Michigan
- Work: UC Berkeley, Michigan, MIT
- Create:
 - INGRES
 - PostGRES
 - SciDB
- Founded 9 startups
- Current startup:
 - paradigm4 (SciDB)
- Famous publication:
[The Red Book \(Readings in Database Systems\)](#)



PostgreSQL

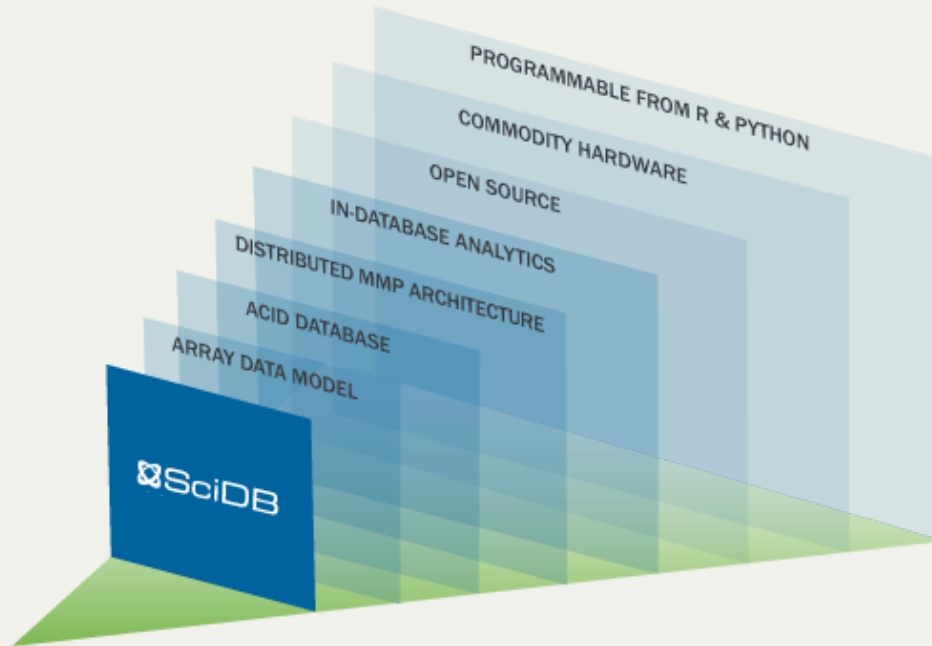
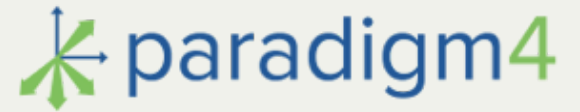


PostgreSQL

Oracle



SciDB



Database types

- Relational DBMS
- Key-value stores
- Document stores
- Graph DBMS
- Wide-column stores
- Search engines
- Time series DBMS

Key-value store

A key-value storage system (or key-value store) is a system that provides a way to store or update a record (value) with an associated key and to retrieve the record with a given key.

Document store

Document store is a semi-structured, document-oriented database designed and optimized to store and work with XML documents such as MongoDB.

Concept of Abstraction

Abstraction allows complexity to be manageable even without full knowledge of the system.

Abstract view of the information enables users and application programmers to administer the database without understanding the full details of how data are stored and organized.

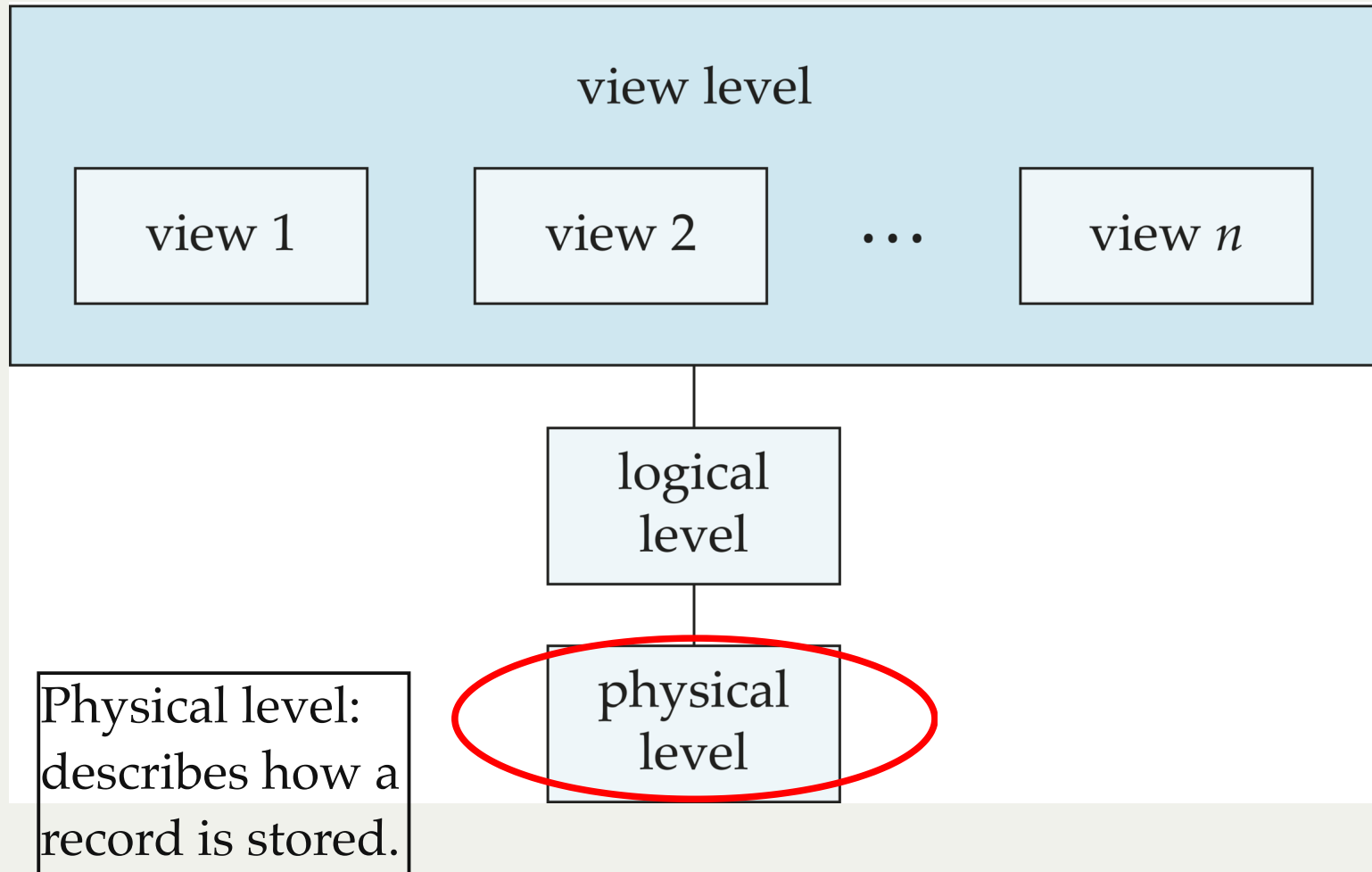
Abstract view of database

A database system is a collection of interrelated data and a set of programs that allow users to access and modify these data. A major purpose of a database system is to provide users with an *abstract* view of the data. That is, the system hides certain details of how the data are stored and maintained.

Abstract view of database

- A major purpose of a database system is to provide users with an abstract view of the data
 - Data models
 - A collection of conceptual tools for describing data, data relationships, data semantics, and consistency constraints.
 - Data abstraction
 - Hide the complexity of data structures to represent data in the database from users through several levels of data abstraction.

Architecture of database



Data model

- a collection of conceptual tools for describing data, data relationships, data semantics, and consistency constraints.
 - Relational Model
 - Entity-Relationship Model
 - Semi-structured Data Model
 - Object-Based Data Model
 - Semi-structured data model (XML)
 - Other older models:
 - Network model
 - Hierarchical model

Data model

A data model specifies the data elements associated with a problem domain, the properties of those data elements, and how those data elements relate to one another.

Data model

In developing a data model, we commonly first identify the entities that are to be modeled and then define their properties and relationships.

For example, the entities include individuals, institutions, parties, each of which has various properties (e.g., for individuals, name, address, employer)

The relationships include “is employed by” and “support” This conceptual data model can then be translated into relational tables or some other database representation.

Data table

A table (also referred to as a relation) is a set of rows (also referred to as tuples, records, or observations), each with the same columns (also referred to as fields, attributes or variables). A database consists of multiple tables.

Database

There are over 300 database system in the market, according to [DB Engines](#). The top three are:

- Oracle
- MySQL
- Microsoft SQL Server

File-processing system

In the early days, database applications were built directly on top of file systems.

What could be the problem for keeping records in files and steel cabinets?



<https://www.databankimx.com/>



<https://history.nasa.gov/computers/Ch8-2.html>

File-processing system

Disadvantages

- Data redundancy and inconsistency
- Difficulty in Data access
- Data isolation
- Integrity
- Atomicity
- Concurrent-access anomalies
- Security

File-processing system

- Data redundancy and inconsistency:
 - data is stored in multiple file formats resulting in duplication of information in different files
- Difficulty in accessing data
 - Need to write a new program to carry out each new task
- Data isolation
 - Multiple files and formats
- Integrity problems
 - Integrity constraints (e.g., account balance > 0) become “buried” in program code rather than being stated explicitly
 - Hard to add new constraints or change existing ones

File-processing system

Atomicity:

- The process must happen in its entirety or not at all.
- e.g. fund transfer from one account to the other account
- Atomicity of updates
 - Failures may leave database in an inconsistent state with partial updates carried out
 - Example: Transfer of funds from one account to another should either complete or not happen at all

File-processing system

- Concurrent access by multiple users
 - Concurrent access needed for performance
 - Uncontrolled concurrent accesses can lead to inconsistencies
 - E.g. Two people reading a balance (say 100) and updating it by withdrawing money (say 50 each) at the same time
- Security problems
 - Hard to provide user access to some, but not all, data

Database Management Systems (DBMS)

A database management system is a software suite designed to safely store and efficiently manage databases, and to assist with the maintenance and discovery of the relationships that database represents.

Database Management Systems (DBMS)

- DBMS contains information about a particular enterprise
 - Collection of interrelated data
 - Set of programs to access the data
 - An environment that is both convenient and efficient to use
- Database systems are used to manage collections of data that are:
 - Highly valuable
 - Relatively large
 - Accessed by multiple users and applications, often at the same time.
- Databases touch all aspects of our lives

File-processing system vs. DBMS

- A database management system coordinates both the physical and the logical access to the data, whereas a file-processing system coordinates only the former
- DBMS reduces the amount of data duplication by ensuring that a physical piece of data is available to all programs authorized to have access to it, whereas file-processing system is for one program only.

File-processing system vs. DBMS

- DBMS allows flexible access to data (i.e., queries), whereas a file-processing system does pre-determined access to data (i.e., compiled programs).
- DBMS can coordinate multiple users accessing the same data at the same time. A file-processing system is usually designed to allow one or more programs to access different data files at the same time. In a file-processing system, a file can be accessed by two programs concurrently only if both programs have read-only access to the file.

Database Management Systems (DBMS)

Three key components:

1. Data model

- defines how data are represented: see Box 4.1)

2. Query language

- defines how the user interacts with the data

3. Transactions and crash recovery

- to ensure reliable execution despite system failures

Database Management Systems (DBMS)

Five responsibilities:

1. interaction with the file manager
2. integrity enforcement
3. security enforcement
4. backup and recovery
5. concurrency control

Key components of DBMS

Table 4.2. Key components of a DBMS

	Data model	Query language	Transactions, crash recovery
User-facing	For example: relational, semi-structured	For example: SQL (for relational), XPath (for semi-structured)	Transactions
Internal	Mapping data to storage systems; creating and maintaining indices	Query optimization and evaluation; consistency	Locking, concurrency control, recovery

Foster et al. 2016, p. 97

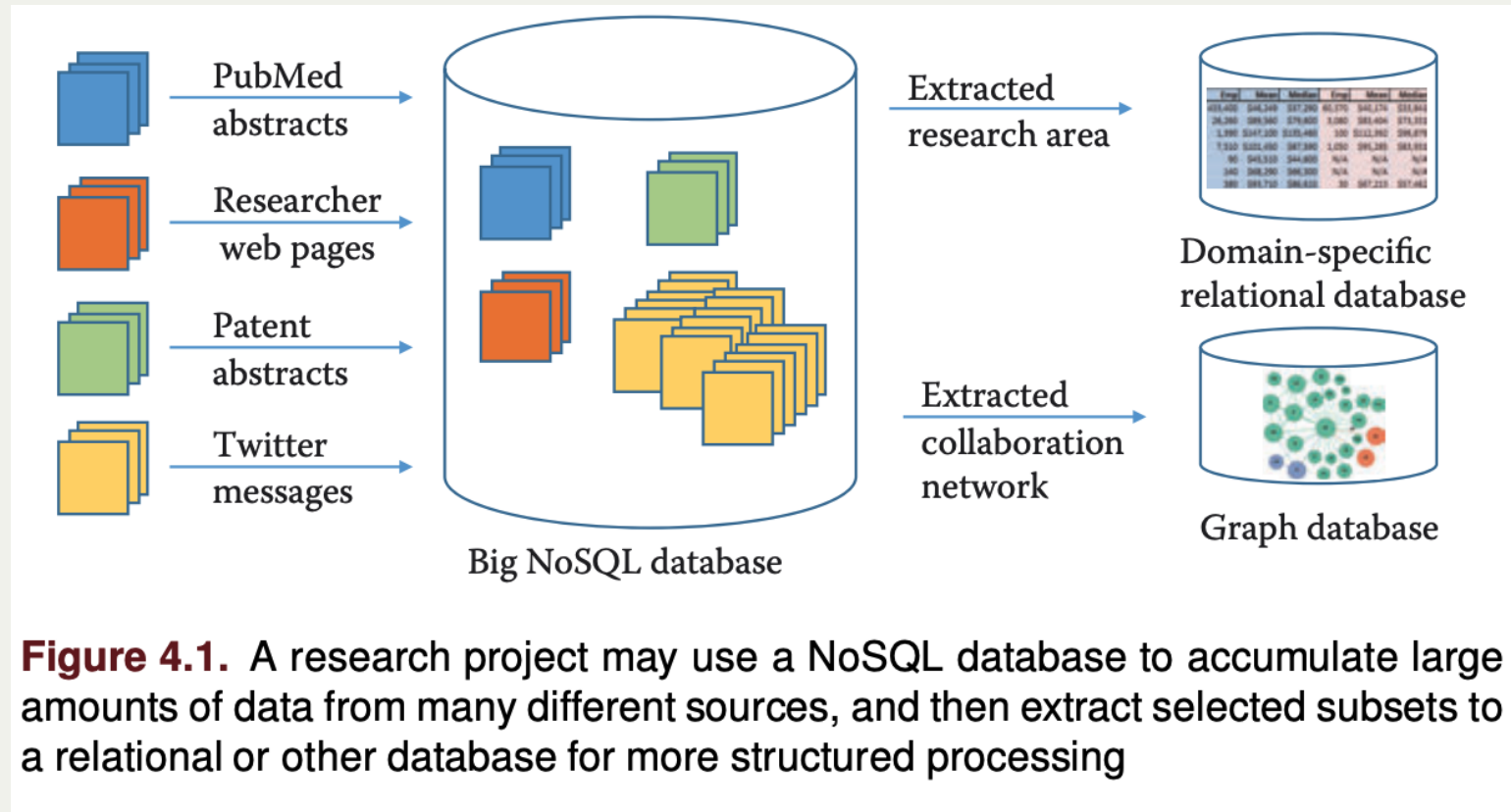
Database types

Table 4.3. Types of databases: relational (first row) and various types of NoSQL (other rows)

Type	Examples	Advantages	Disadvantages	Uses
Relational database	MySQL, PostgreSQL, Oracle, SQL Server, Teradata	Consistency (ACID)	Fixed schema; typically harder to scale	Transactional systems: order processing, retail, hospitals, etc.
Key-value store	Dynamo, Redis	Dynamic schema; easy scaling; high throughput	Not immediately consistent; no higher-level queries	Web applications
Column store	Cassandra, HBase	Same as key-value; distributed; better compression at column level	Not immediately consistent; using all columns is inefficient	Large-scale analysis
Document store	CouchDB, MongoDB	Index entire document (JSON)	Not immediately consistent; no higher-level queries	Web applications
Graph database	Neo4j, InfiniteGraph	Graph queries are fast	Difficult to do non-graph analysis	Recommendation systems, networks, routing

Foster et al. 2016, p. 99

Relational Database and NoSQL



Foster et al. 2016, p. 100

NoSQL

- document-oriented database
- key-value database
- column-oriented database
- graph database
- Hadoop system

What's new?

PostgreSQL

- PostgreSQL is an open-source object-relational database management system, first developed by Michael Stonebraker at the University of California, Berkeley.
- The name “*postgres*” is derived from the name of a pioneering relational database system, *Ingres*, also developed by Stonebraker.
- PostgreSQL supports many aspects of SQL:2003 and offers features such as complex queries, foreign keys, triggers, views, transactional integrity, full-text searching, and limited data replication.
- Extensible with new data types, functions, operators, or index methods.
- PostgreSQL supports a variety of programming languages (including C, C++, Java, Perl, Tcl, and Python) as well as the database interfaces JDBC and ODBC.

Cassandra

Cassandra is an open-source, distributed NoSQL database management system by Apache.

The wide column store is designed to handle large amounts of data across many commodity servers. It is built with Java so it is cross-platform.

Database design tips:

When designing a data structure, we have to think about how to manage its growth and the possible implications of the chosen technique.

A sample relational database

<i>ID</i>	<i>name</i>	<i>dept_name</i>	<i>salary</i>
22222	Einstein	Physics	95000
12121	Wu	Finance	90000
32343	El Said	History	50000
45565	Katz	Comp. Sci.	75000
98345	Kim	Elec. Eng.	80000
76766	Crick	Biology	72000
10101	Srinivasan	Comp. Sci.	65000
58583	Califieri	History	62000
83821	Brandt	Comp. Sci.	92000
15151	Mozart	Music	40000
33456	Gold	Physics	87000
76543	Singh	Finance	80000

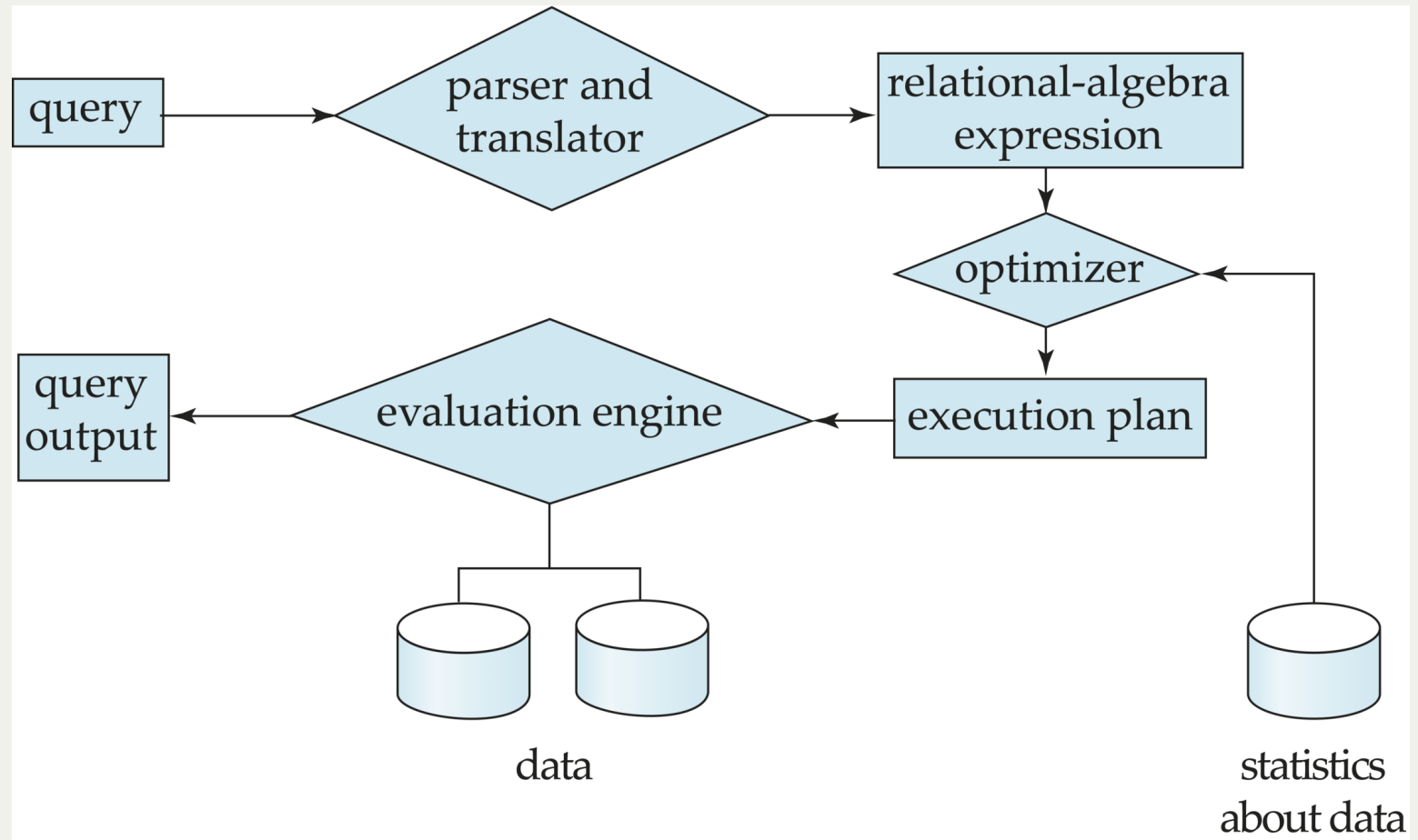
(a) The *instructor* table

<i>dept_name</i>	<i>building</i>	<i>budget</i>
Comp. Sci.	Taylor	100000
Biology	Watson	90000
Elec. Eng.	Taylor	85000
Music	Packard	80000
Finance	Painter	120000
History	Painter	50000
Physics	Watson	70000

(b) The *department* table

SKS, p. 9

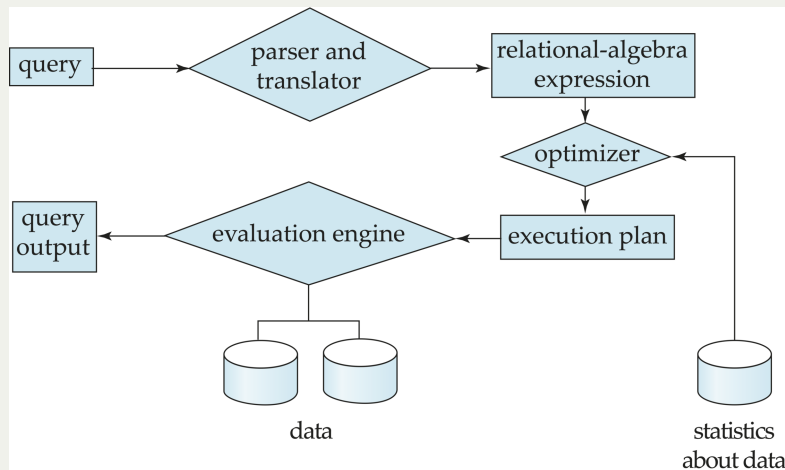
Query processing



Query processing

Query processing refers to the range of activities involved in extracting data from a database including:

- translation of queries in high-level database languages into expressions that can be used at the physical level of the file system
- variety of query-optimizing transformations
- actual evaluation of queries.



Steps in query processing:

- Parsing and translation.
- Optimization.
- Evaluation.