Database systems

Jaroslav Porubän, Miroslav Biňas,
Milan Nosáľ (c) 2011 - 2016

Resources

- Ramez Elmasri, Shamkant B. Navathe: *Fundamentals of Database Systems*, Addison Wesley, 5 edition, 2006, 1168 p. ISBN 0321369572.
- S. Sumathi, S. Esakkirajan: *Fundamentals of Relational Database Management Systems*, Springer, 2007, 776 p. ISBN 3540483977.
- Abraham Silberschatz, Henry F. Korth, S. Sudarshan: *Database System Concepts*, The McGraw-Hill Companies, 2011, 6th edition, ISBN 978-0-07-352332-3.

Resources

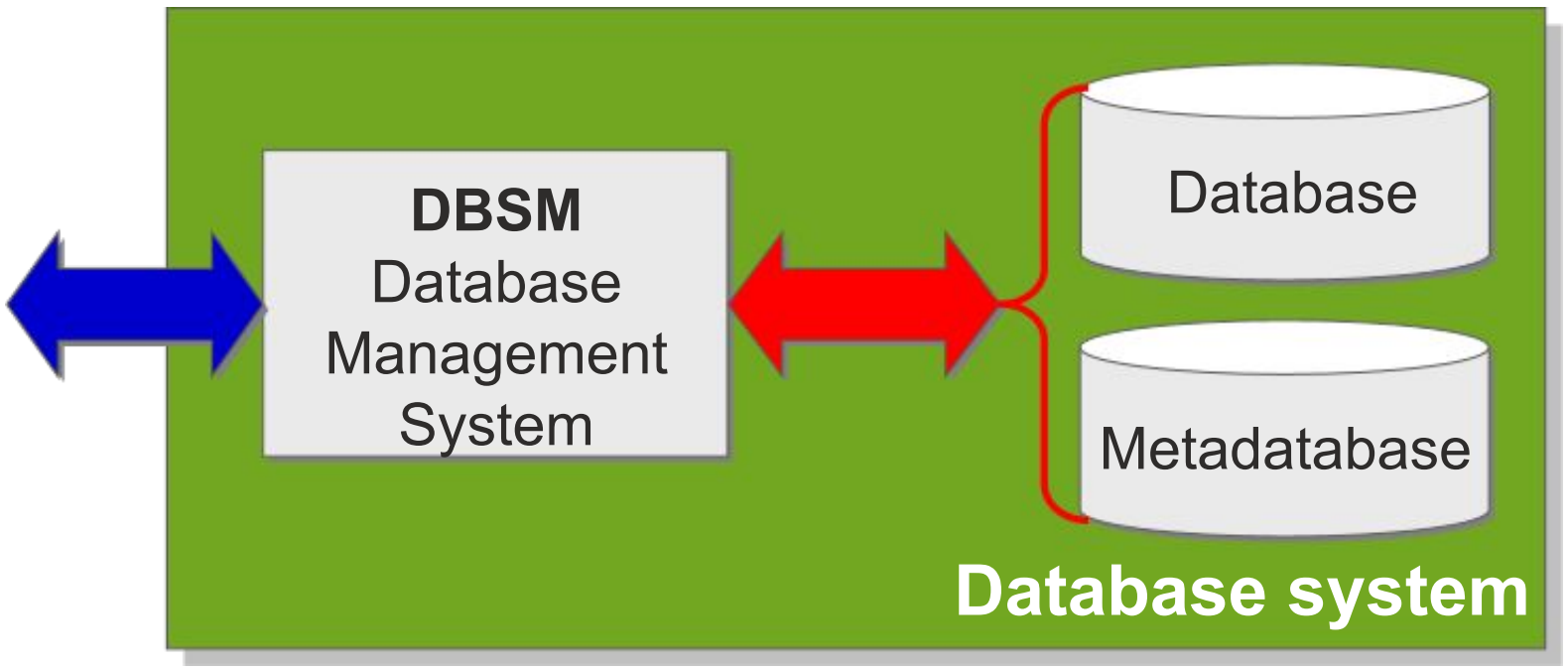
- Jennifer Widom: *Databases*, Stanford University, Stanford, California 94305
  - https://lagunita.stanford.edu/courses/Home/Databases/Engineering/about
- Database systems vendors:
  - https://www.mysql.com/
  - http://www.oracle.com/
- Wikipedia

Database

- Database is an **organized collection of related data** (facts that can be recorded and have some implicit meaning)
- Database represents some **aspect of real world - miniworld** (or universe of discourse)
  - E.g., records data about employees
- Database is designed, build, and populated with data for a **specific purpose**, which is defined by the character od recorded data

Database system

- **Database system DBS = DBMS + DB**
  - **DBMS** – Database Management System
    - General usage (different domains)
  - **DB** – database



Storing data in files I.

- Data redundancy and inconsistency
  - The same piece of data is recorded multiple times, and its instances has to be synchronized
- Complicated access to data
  - New program for each problem, different formats for storing data
- Data isolation
  - Data are scattered in multiple files
- Integrity issues
  - Contraints for data integrity are hidden in program source code

Storing data in files II.

- Changes atomicity
  - System failure during data update leaves data inconsistent
- Concurrent access to data
  - Necessary for performance
  - Uncontrolled concurrency can cause data inconsistency
- Security
  - User and roles management

DBMS purpose I.

- **Data consistency and irredundancy**
  - Stored data are a reliable source, and they are stored only once (backup does not count)
- **Data accessibility**
  - Data are easily accessible to multiple different users in a comprehensible form
  - User does not manage files
  - High level query languages are supported
- **Data integrity**
  - Database system contains also database description, that is managed by DBMS

DBMS purpose II.

- **Data persistence**
  - Stored data will survive restart of application working with them
- **Data safety**
  - Fault tolerance (hardware, software, user)
  - Transaction-based processing
  - Database restore and recovery
- **Multi-user environment**
  - Supports concurrent access of large numbers of users (e.g., tousands)
  - Controlled access to data based on roles and users

DBMS purpose III.

- **Data and program independence**
  - Data are presented in form independent from the format used to store them
  - Data can be shared for multiple purposes (as well as programs)
- **Performance**
  - One of the most important characteristics
  - Efficient processing of massive data (tera- or even petabytes)

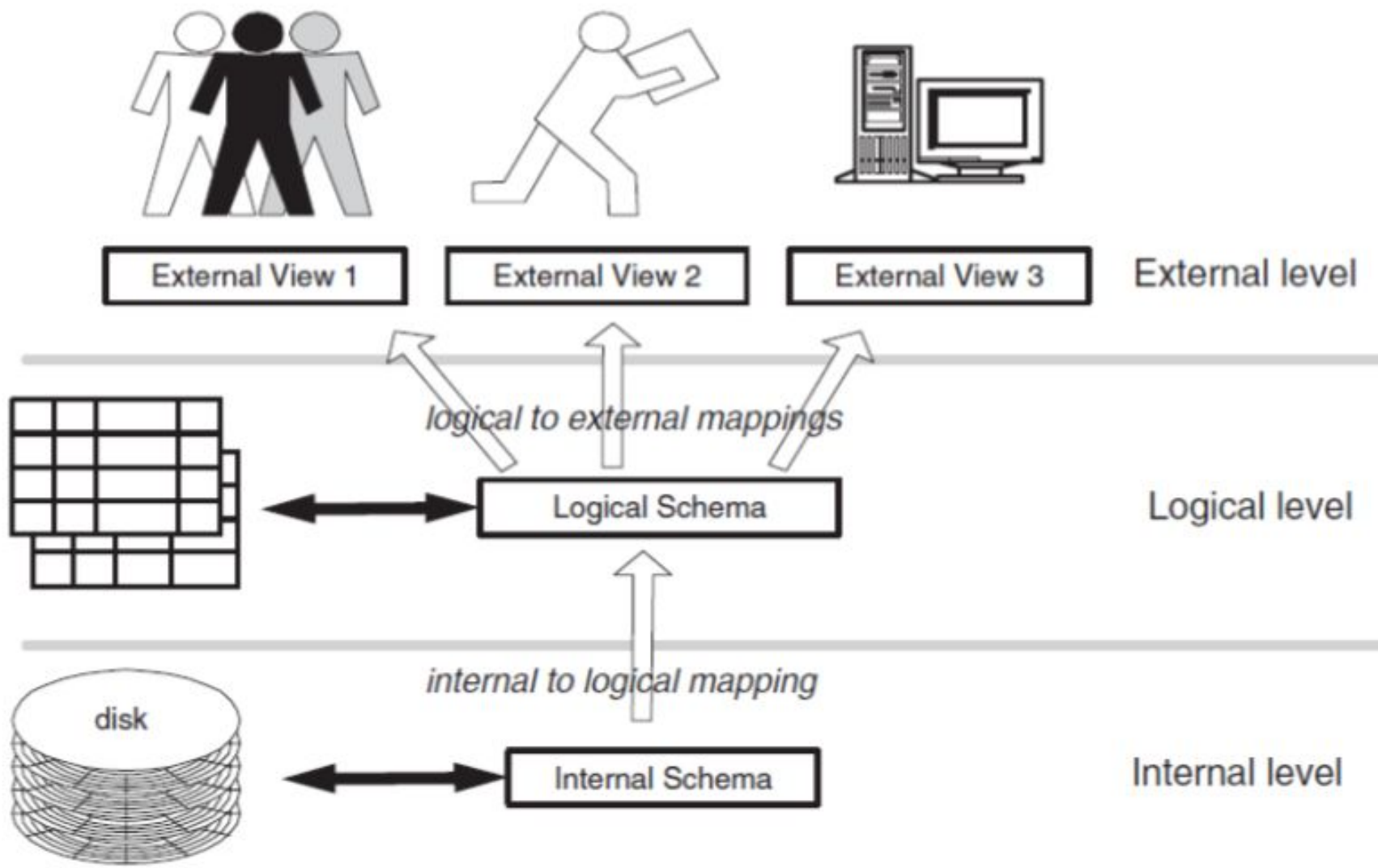
Data abstraction and independence I.

- **Database schema** – database design and its structure
  - **Internal schema** – describes details about physical data storage - physical format
  - **Conceptual schema** – describes logical structure of the data and data relations - implementation model
  - **External (view) schema (subschema)** - user view of database (program) - there can be multiple different views to the same database

Data abstraction and independence II.

- **Mapping** - process of transforming requests between different schemas (levels)
  - **Data independence** - change of schema on one level does not require change on higher level

DB Views


External View 1
External View 2
External View 3
External level
logical to external mappings
Logical Schema
Logical level
internal to logical mapping
disk
Internal Schema
Internal level

Data model

- Collection of tools for describing data structure on logical level = **database structure**
  - Data
  - Data relationships
  - Data consistency constraints
- Data models
  - Conceptual (entity-relationship model)
  - Implementation (relational model v RDBMS)
- Other implementation models
  - graph, hierarchical, key-value pair, etc.

DBS users I.

- **Database Administrator**
  - Grants access to database
  - Coordinates and monitors usage
  - Manages hardware and software resources
  - Manages backup and recovery

DBS users II.

- **Database Designer**
- **Database User**
  - **Application developers**
  - Sophisticated user
    - Uses directly query language
  - Specialized user
    - Sophisticated user developing applications for a specific purpose
  - Naive user
    - User of applications that use DBS

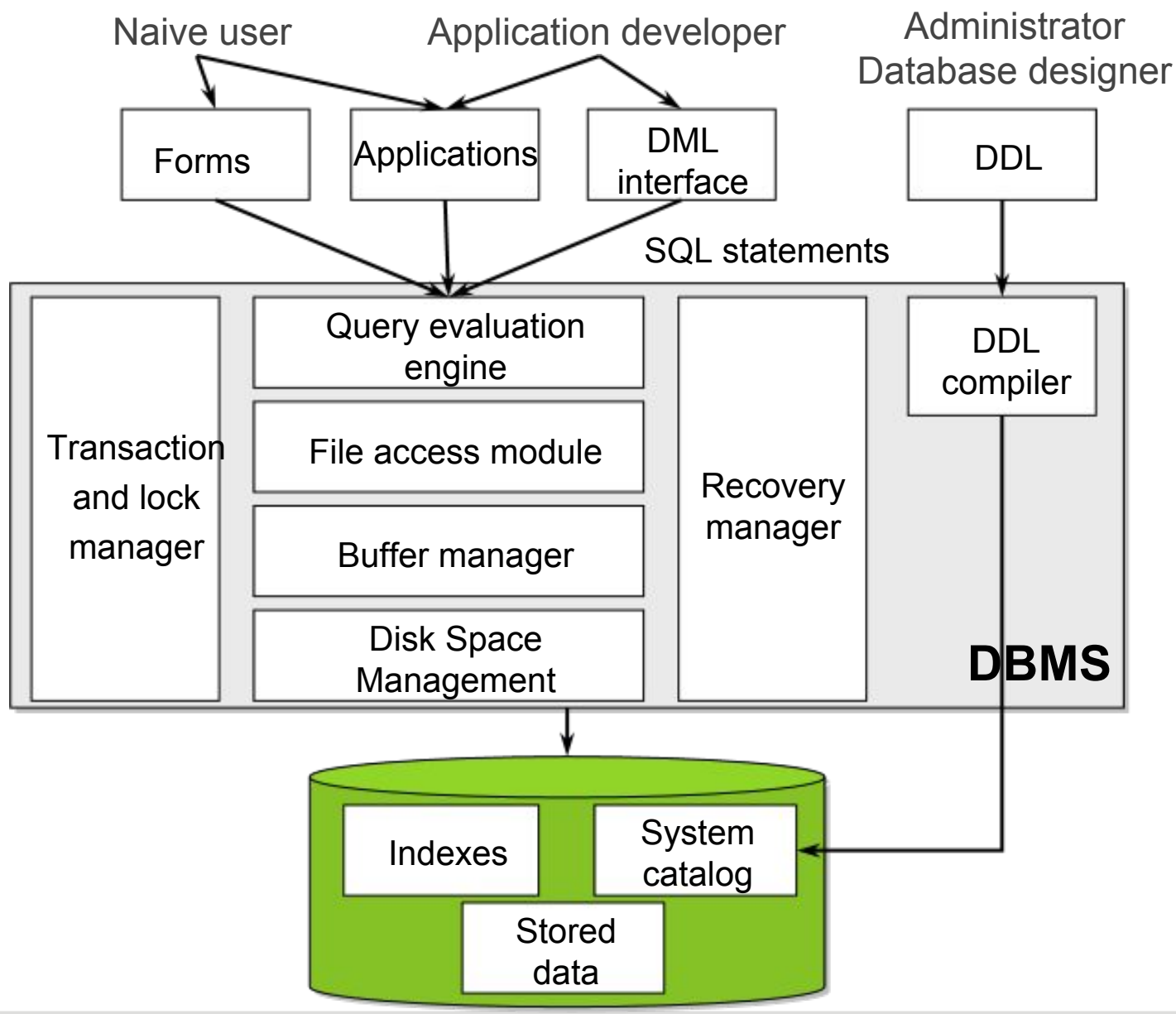
Data dictionary

- **Data dictionary**, or **system catalog**
  - Centralized storage of information of the database
  - Data stored in system catalog are metadata
- **Metadata**
  - Data about data – describe structure of data

DBMS components

- **Data**
  - persistent
  - integrated
  - structured (well organized)
  - shared(multiple users/applications in real time)
- **Hardware**
- **Software**
- **Methods**
  - Rules for database design and usage (start/stop, restore db, ...)
- **Users**

Simplified DBS Architecture

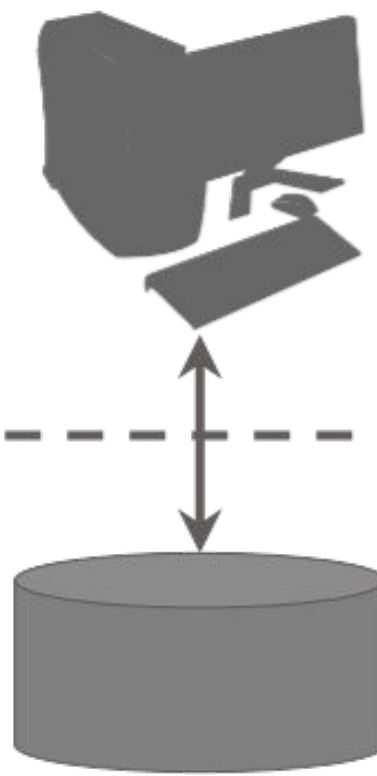


Database application architecture

- **Application services (purposes)**
  - User interface
  - Presentation services
  - Bussiness + application services
    - Domain
  - Data
- Two-tier architecture
- Three-tier architecture

Two-tier architecture

First tier
Client

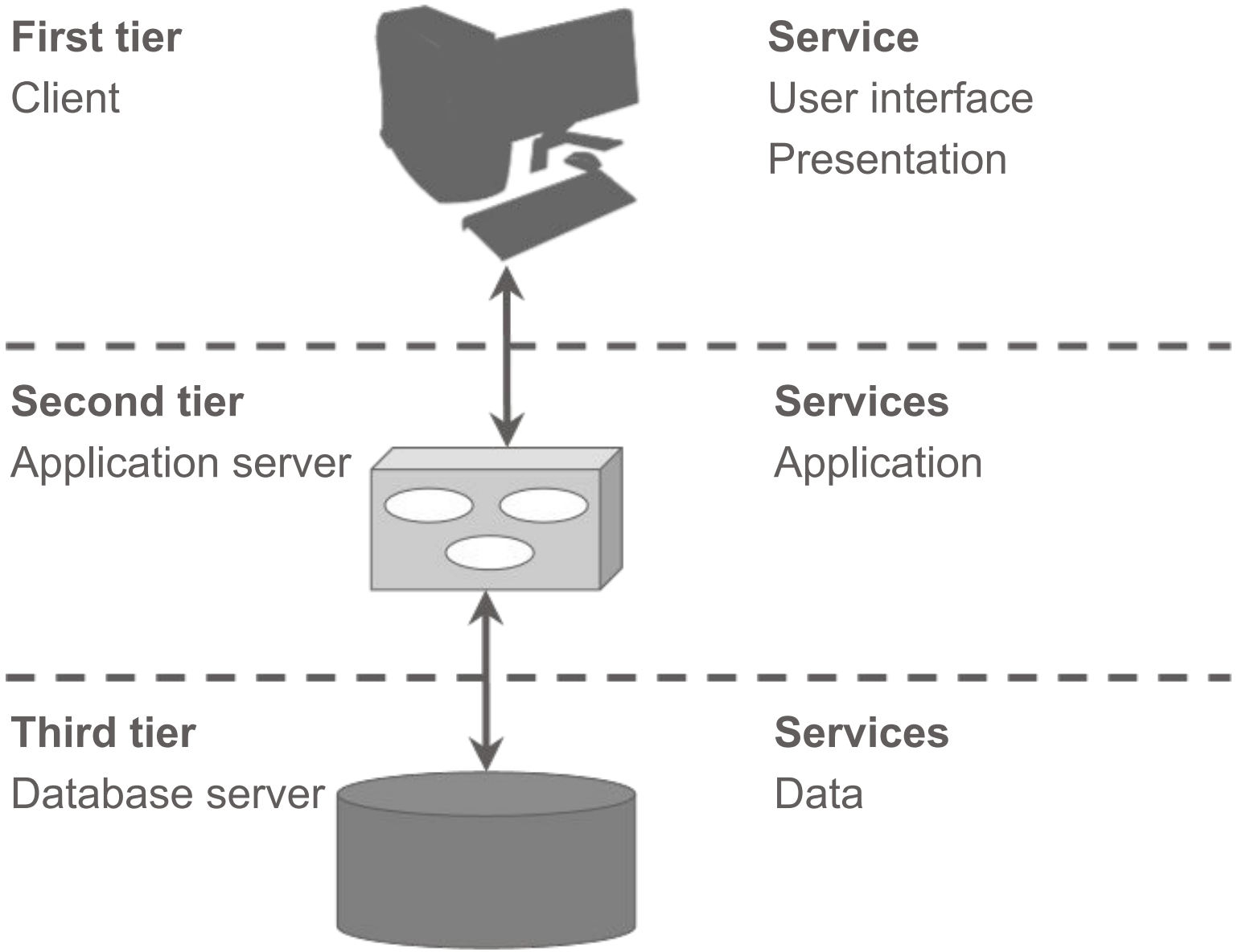


Services
User interface
Presentation
Application

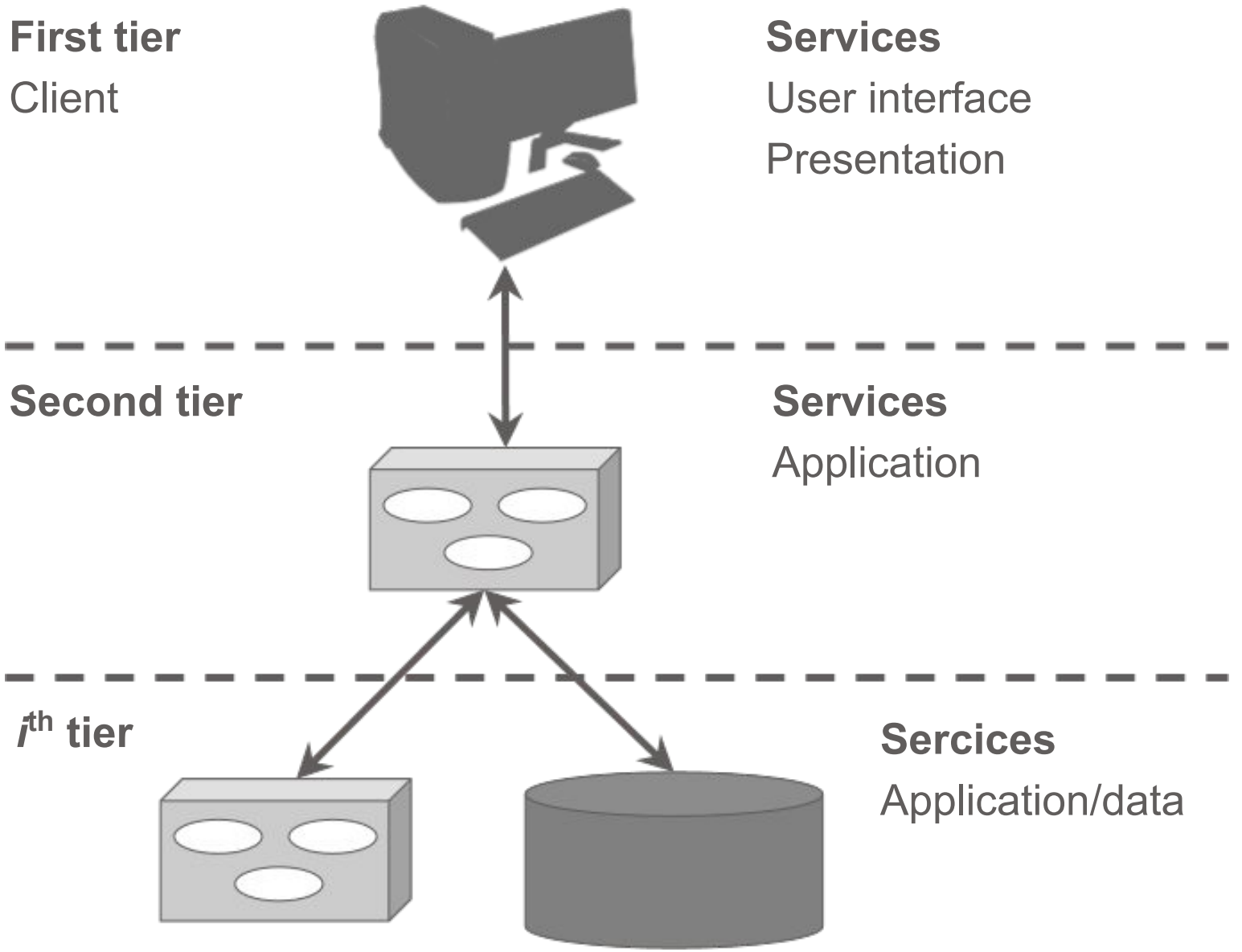
Second tier
Database
server

Services
Application
Data

Three-tier architecture



Multi-tier application



Questions?