Data Integration

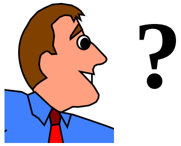
Knowledge Objectives

1. Identify the problem of information integration
2. Enumerate the three solutions to information integration
3. Explain the three characteristics of a distributed database
4. Distinguish the five kinds of distributed systems
5. Name five kinds of system heterogeneities
6. Name four kinds of semantic heterogeneities on instances
7. Name five kinds of semantic heterogeneities on classes
8. Name four kinds of structural semantic heterogeneities along generalization/specialization
9. Name four kinds of structural semantic heterogeneities along aggregation/decomposition
10. Explain what a wrapper-mediator architecture is

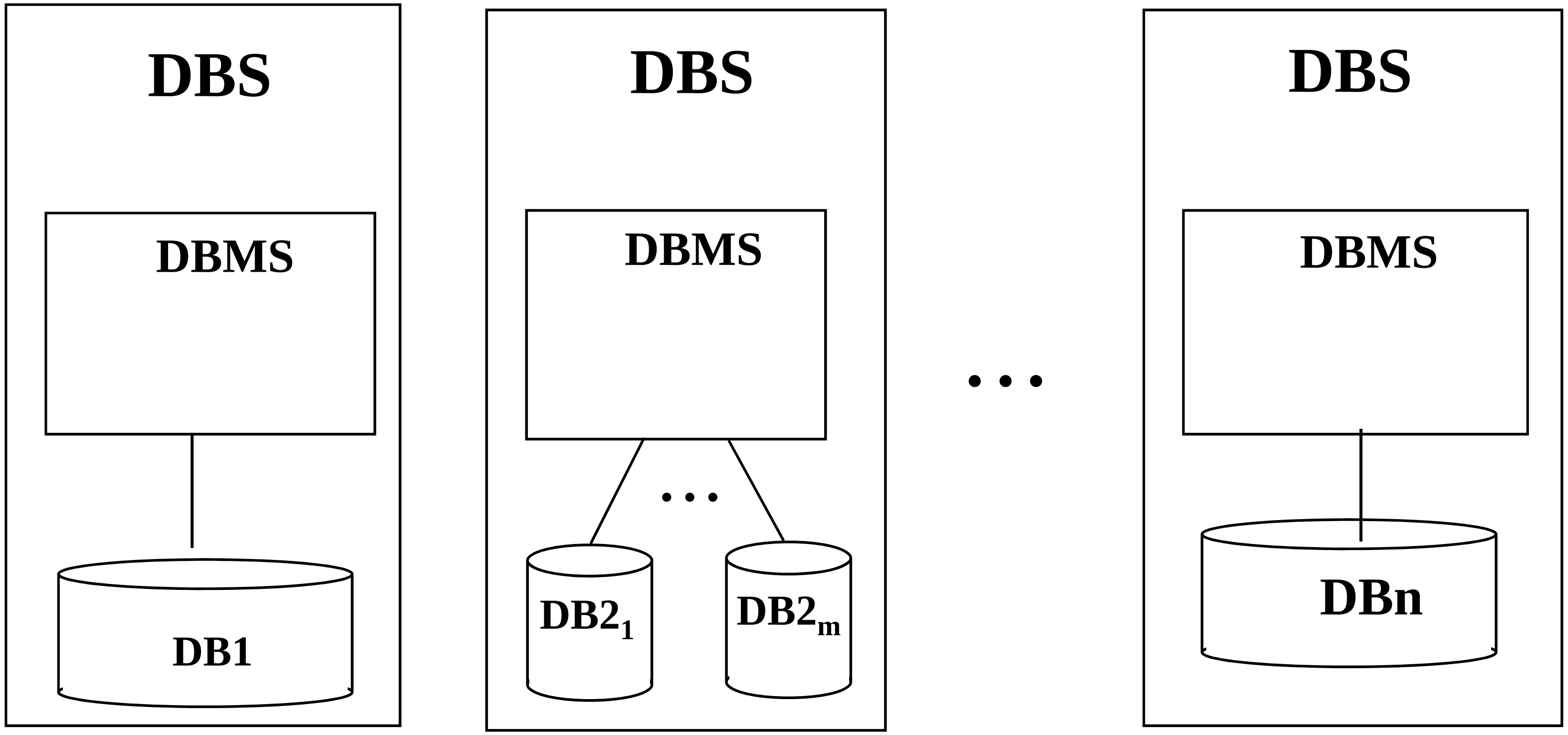
Application Objectives

1. Given two schemas of the same domain with structural discrepancies, decide whether they represent the same reality or not
2. Given a schema, propose an equivalent alternative showing any kind of semantic heterogeneity

The problem (I)



Answer a query that requires accessing several databases



The problem (II)

Being able to pose **one query**, and get **one answer**, so that in the preparation of the answer data coming from **several DBs** is processed.

It is not:

- BD connectivity (assumed)
- Electronic Data Interchange (EDI), Business to Business (B2B), eB-XML (p.ej. SOAP)
- Remote database access
- Multiclient/Multiserver architecture
- Distributed DBMS

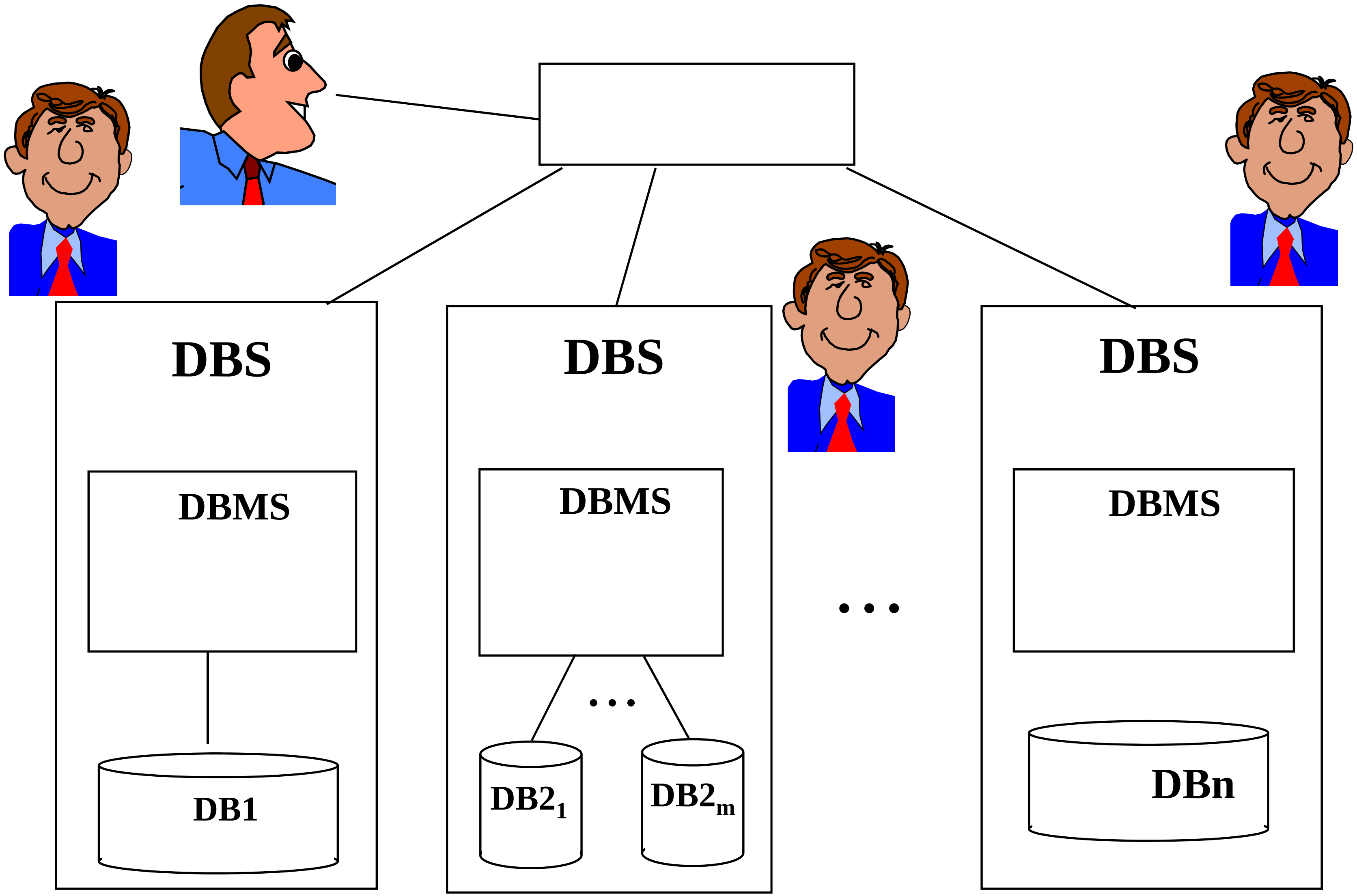
Solutions

a) Manually query the different databases separately
- Know the available databases
  - Data
  - Data model
    - Query language
- Decompose the query
- Integrate the results

b) Create a new database containing all necessary data
- Design it
- Move data
- Modify the applications to use the new repository
- Test everything

c) Build a software layer on top of the databases that automatically splits the queries and integrates the answers
- Add a new software layer that defines two access levels
- Automatically process the queries

Users in the integrated system



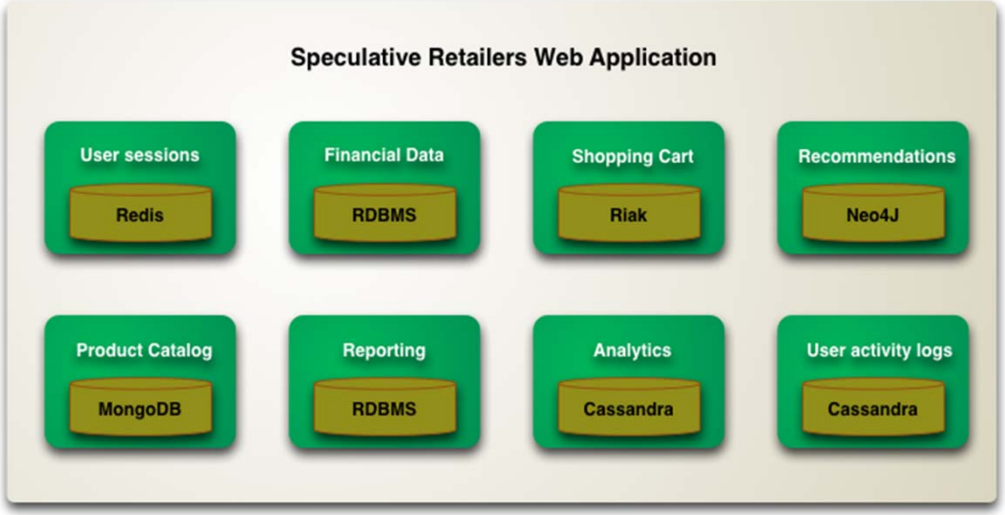
Distributed Database

"A distributed database (DDB) is a collection of multiple, **logically interrelated** databases (known as nodes or sites) **distributed over a computer network**. A distributed database management system (DDBMS) is thus, the software system that permits the management of the distributed database and makes the **distribution transparent to the users**."

Tamer Özsu & P. Valduriez
Principles of DDB Systems
Springer, 2011

Classification of DDB

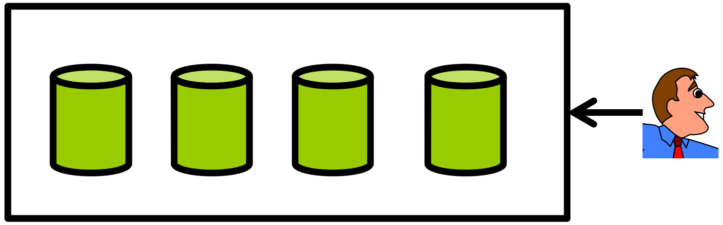
- Autonomy
  a) Design
  b) Execution
  c) Association
- Heterogeneity
  a) System
  b) Semantic



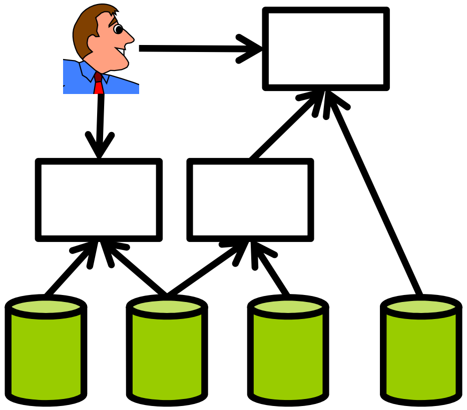
Polyglot persistence, Martin Fowler

Kinds of heterogeneous systems

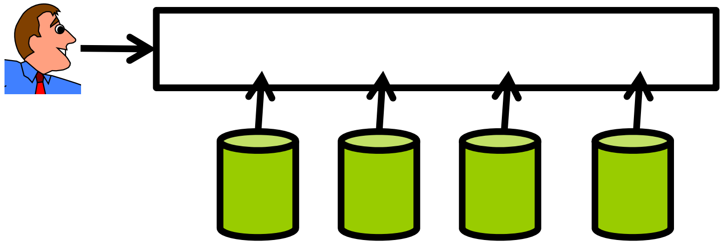
DDBMS



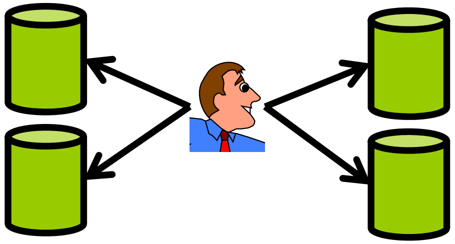
Mediators



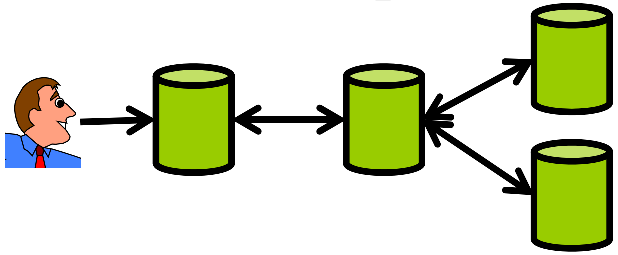
Federated



Multi-database



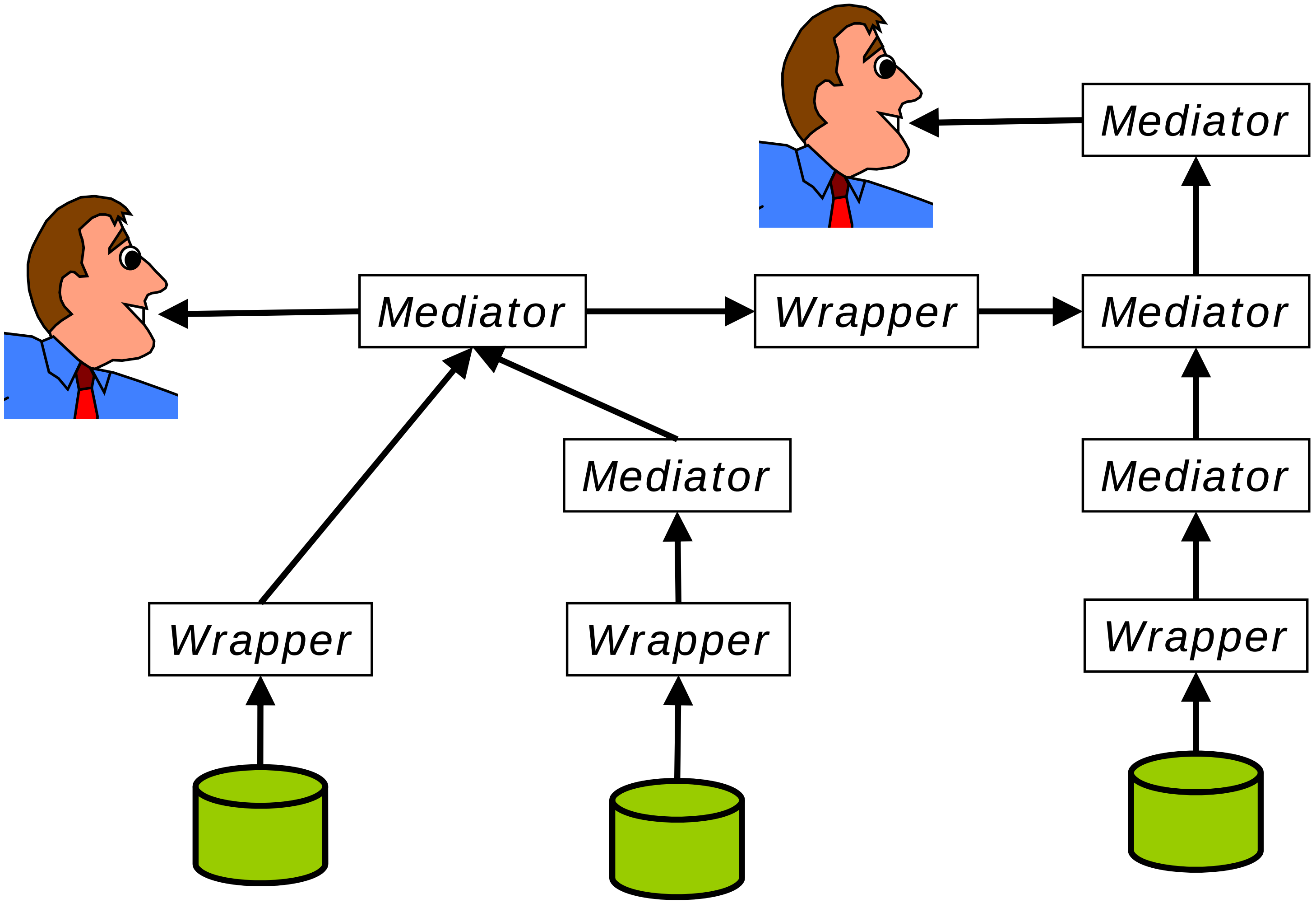
Peer-to-peer



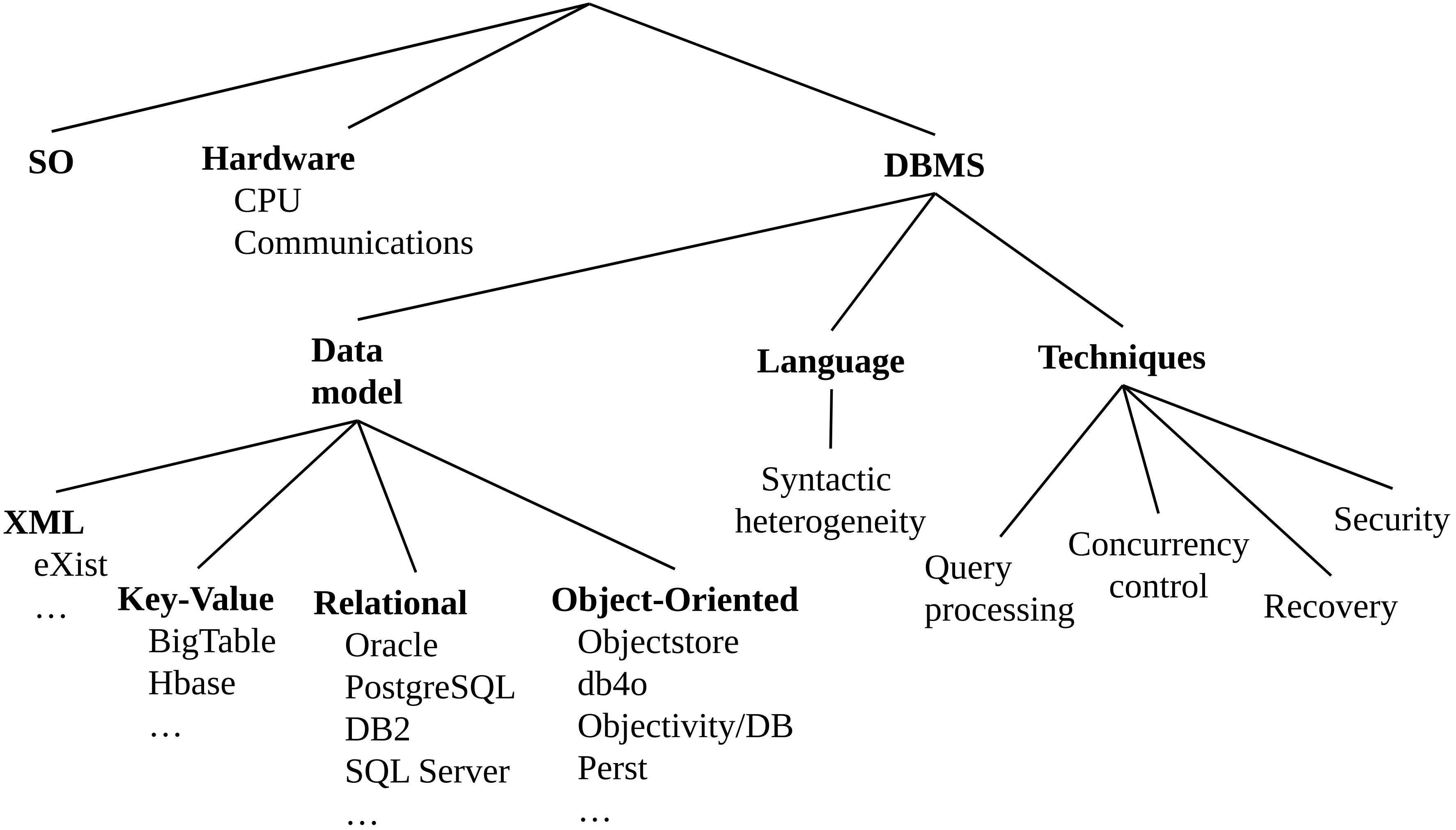
Comparison of heterogeneous systems

	Autonomy	Central schema	Query transparency	Query consistency	Update transparency
DDBMS	No	Yes	Yes	Yes	Yes
Federated	Yes	Yes	Yes	Yes	Limited
Mediators	Yes	No	Yes	Yes	Limited
Multi-database	Yes	No	No	Yes	No
Peer-to-Peer	Yes	No	Yes	No	No

Wrapper-Mediator architecture



System heterogeneities


SO

Hardware
CPU
Communications

DBMS

Data model

XML
eXist
…

Key-Value
BigTable
Hbase
…

Relational
Oracle
PostgreSQL
DB2
SQL Server
…

Object-Oriented
Objectstore
db4o
Objectivity/DB
Perst
…

Language

Syntactic heterogeneity

Techniques

Query processing

Concurrency control

Recovery

Security

Semantic heterogeneities: Instances

- Presence/Absence
- Number of values (multi/mono-valued)
- Existence of null values
- Value

Semantic heterogeneities: Classes

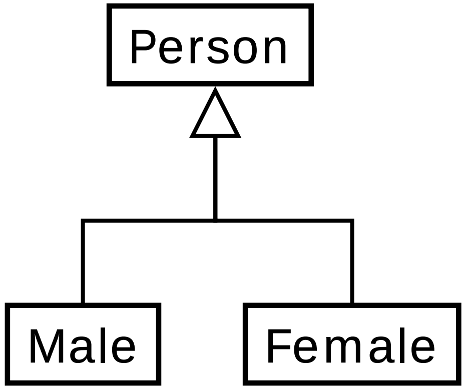
- Extension (e.g., coding colors)
- Name
- Attributes/Methods
  - Presence/Absence
  - Arity
  - Integrity constraints (e.g., mono/multi-valued)
- Domain
  - Keys
  - Data types
  - Dimension (e.g., volume vs weight)
  - Measuring units (e.g., liters vs gallons)
  - Scale (e.g., liters vs m^3)
- Constraints (checks and assertions)

Semantic heterogeneity: Structure

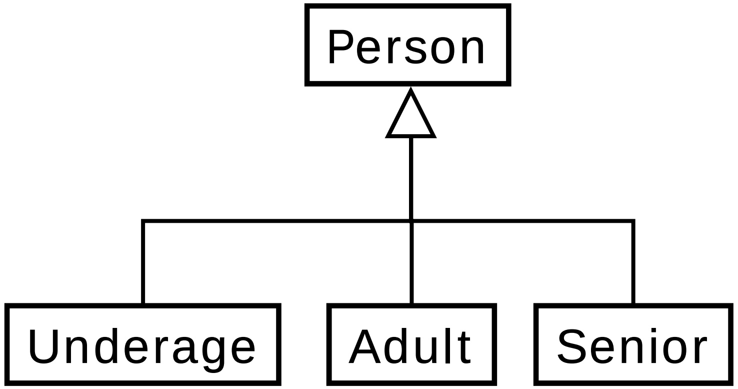
- Generalization/Specialization
  - Criterion (e.g., sex vs job)
  - Degree and characterization (e.g., different groups of age)
  - Kind (i.e., complete or not, disjoint or overlapping)
  - Integrity constraints (e.g., delete effect)
- Aggregation/Decomposition
  - Kind of aggregation (i.e., composition or not)
  - Participating classes
    - Specialization in the aggregated class (e.g., parent vs father)
    - Collection in the aggregated class (e.g., projects vs subprojects)
    - Composition in the aggregated class (e.g., address vs street+number+city)
  - Kind of partitioning collection (i.e., complete or not, disjoint or overlapping)
  - Component class of the collection (e.g., collection of counties vs collection of states)
- Schematic
  - Specialization vs Composition
  - Data vs Metadata

Example of specialization discrepancies

Option 1

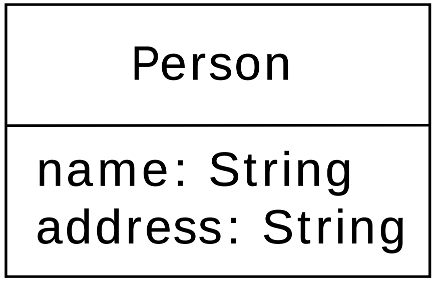


Option 2



Example of aggregation discrepancies (I)

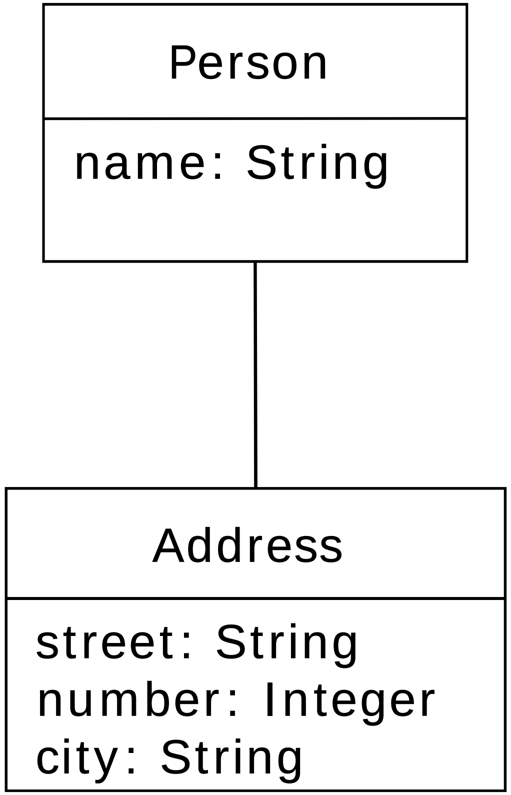
Option 1



Person

name: String
address: String

Option 2



Person

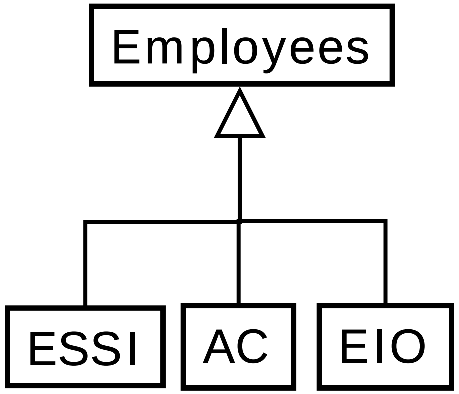
name: String

Address

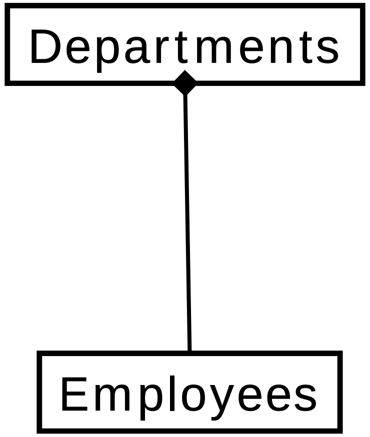
street: String
number: Integer
city: String

Example of schematic discrepancies (I)

Option 1



Option 2



Example of schematic discrepancies (II)

Option 1

Mothers

child: Person
mother: Person

Fathers

child: Person
father: Person

Option 2

Parenthood

child: Person
father: Person
mother: Person

Option 3

Parenthood

child: Person
parent: Person
kind: {Father, Mother}

Summary

- Distributed databases classification
- Heterogeneities
  - System
  - Semantic

Bibliography

- T. Özsu and P. Valduriez. *Principles of Distributed Database Systems*. Springer, 2011
- O. A. Bukhres and A. K. Elmagarmid (Eds.). *Object-Oriented Multidatabase Systems*. Prentice-Hall, 1996
- H. Garcia-Molina et al. *Database Systems*. Prentice Hall, 2009