



Weaknesses of RDBMSs

Representation of 'real world' entities: The process of normalisation generally leads to the creation of relations that do not correspond to entities in the 'real world'.

Semantic overloading: The relational model has only one construct for representing data and data relationships: the relation.

Homogeneous data: The relational model assumes both horizontal and vertical homogeneity. Also, intersection of a row and column must be an atomic value => this structure is restrictive for many 'real world' objects with a complex structure.

3

8

Weaknesses of RDBMSs (cont.) Limited operations: The relational model has a fixed set of operations (provided in SQL). => does not allow new operations to be specified. Recursive queries: It is extremely difficult to produce recursive queries (queries about relationships that a relation has with itself). Impedance mismatch: Result of mixing different programming paradigms (e.g., SQL is a declarative language that handles rows of data whereas a high-level language such as 'C' is a procedural language that can handle only one row at a time).

Object-Oriented Data Model

//Loosely speaking, and *object* corresponds to an entity in the E-R model.

*//*The *object-oriented paradigm* is based on *encapsulating* code and data related to an object into a single unit.

//The object-oriented data model is a logical model (like the E-R model).

//Adaptation of the object-oriented programming paradigm (e.g., Smalltalk, C++) to the database systems.

5

Ś























Class hierarchy Example (cont.) //Full variable list for objects in the class officer: - office-number, expense-account-number: defined locally

- start-date, salary: inherited from employee
- name, address: inherited from person
- //Methods inherited similar to variables.
- *Substitutability* any method of a class, say *person*, can be invoked equally well with any object belonging to any subclass, such as subclass *officer* of *person*.

//class extent: set of all objects in the class. Two options:

- 1 Class extent of *employee* includes all *officer*, *teller* and *secretary* objects
- 2 Class extent of *employee* includes only employee objects that are not in a subclass such as *officer*, *teller* or *secretary*















