

Class Hierarchies

- Entities in a set may have a natural “inheritance” relationship
 - i.e. Two entity sets E_1, E_2 ; $x \in E_1 \rightarrow x \in E_2$
 - “IsA” inheritance
 - Entity x should have all attributes that entities in E_2 have, and perhaps more
- The DBMS must know to look at entity set E_1 when asked a question about E_2

Copyright © Ben Carter@Ge

23

Class Hierarchies

- Hierarchies model **generalization** and **specification**
 - Specification: the highest-level class specializes into multiple lower-level classes with distinct characteristics
 - Generalization: lower-level classes have common characteristics that can define a higher-level class
- Hierarchical constraints:
 - **Overlap constraints:** Can two subclasses contain the same entity? If so, they *overlap*
 - **Completeness constraints:** Does every entity have a subclass? If so, hierarchical relationship is *total*

Copyright © Ben Carter@Ge

24

Aggregation

- We may want to model a relationship between entities and another relationship
- **Aggregation** indicates that a relationship set participates in another relationship set
 - Why a separate feature? Because relationships are defined in terms of entities only
 - An example of a higher-level decision influencing lower-level decisions
 - Why not redefine relationships to include other relationships? Then relationships need keys

Copyright © Ben Carter@De

25

Summary

- The first step in designing a database and database application is **requirements analysis**
- Requirements analysis leads to a **model** of the data
- The **entity-relationship model** is a simple but flexible way to model data in terms of **entities** and **relationships**
 - ER diagrams allow easy visualization of the model

Copyright © Ben Carter@De

26