MODELING TIPS

Modeling Tips

- Entity vs. Attribute
- Atomic Attributes
- Foreign Keys vs. Relationships
- Code Entities
- Store vs. Compute
- Choosing Primary Keys

Atomic Attributes

Worse

- If an attribute contains multiple pieces of data, split it into separate attributes

Alumnus
alumnus_id
full_name
address

Better

Alumnus
alumnus_id
first_name
last_name
street_addr
city
state
zip

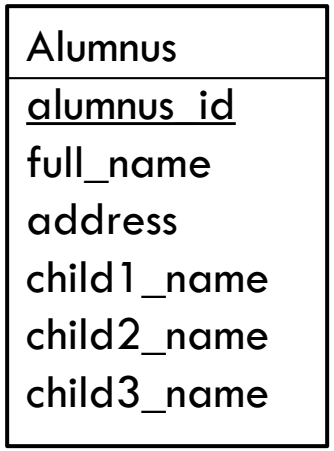
Entity vs. Attribute

- Does it have attributes?
  - It's an entity
- Is it a simple value in real life?
  - Text, number, date
  - Probably attribute
- Is it an attribute that has more than one value?
  - Need an entity

Handling Multi-Valued Attributes

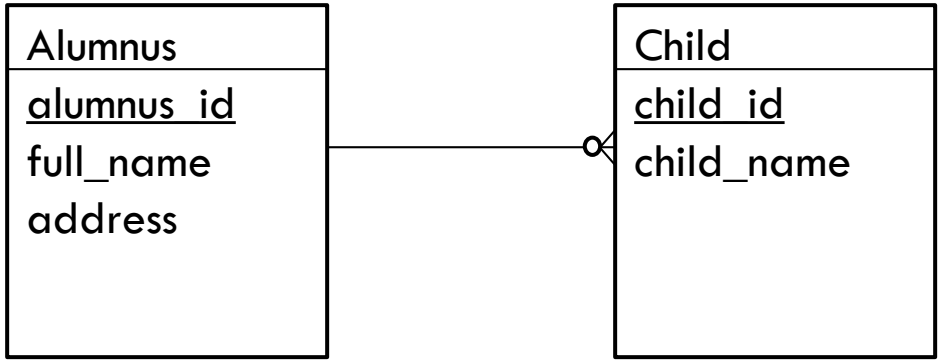
Worse

- Avoid multiple copies of an attribute



Better

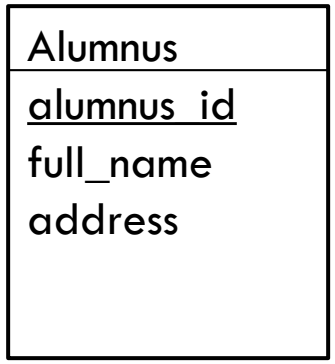
- Create a separate entity for multi-valued attributes

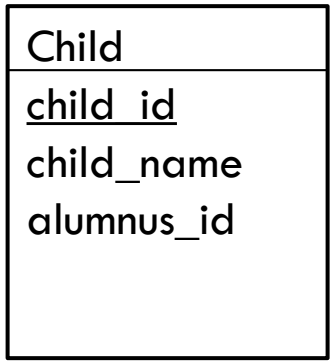


Foreign Key vs. Relationship

Worse

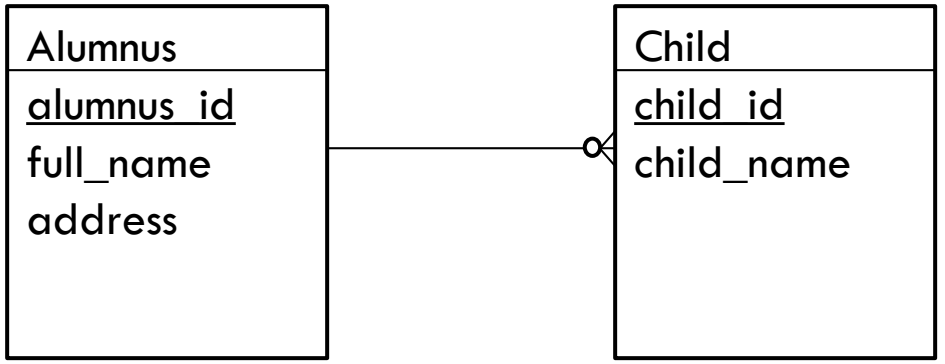
- Don't show foreign keys as attributes





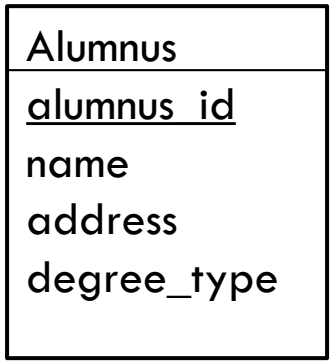
Better

- Draw relationships instead

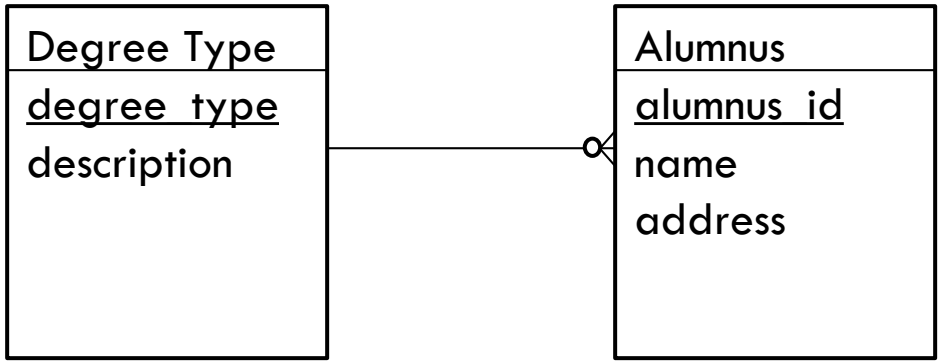


Code Entities

- Some attributes have a small set of permissible values
  - Example: Alumnus degree (bachelor's, master's, ph.d.)
- Create a separate "code entity" to
  - define the legal values (ex. 0 – bachelor's, 1 – master's, 2 – ph.d.)
  - provide display values for reports



Without code entity



With code entity

Store vs. Compute

- Consider this schema:
  - Order(order_id, order_date, customer_id, order_total)
  - Order_Line(line_id, order_id, product_id, quantity, ext_price)
  - Product(product_id, description, price)
- Are any of these attributes computable from other attributes?
- In general:
  - Information that can be computed from other attributes should not be stored as an attribute

Store vs. Compute

- A computable attribute should be stored as an attribute in the model if:
  - User must be able to manually override the computed value
  - Need to preserve originally computed value if values on which it depends change
  - Value is expensive to compute

Primary Key

Pick a column that is

- Unique
- Stable
- Small
- Used only for entity and referential integrity