

WEAK ENTITY-SET RULES

- A weak entity set has one or more many-one relationships to other (supporting) entity sets.
-
- is guaranteed).

53

WEAK ENTITY-SET RULES

- The key for a weak entity set is its own underlined attributes and the keys from supporting entity sets.

e.g. number (Players) and name (Teams) is a key for **Players**.

54

EXAMPLE

```

    erDiagram
        Players ||--o{ Teams : Plays-on
        Players {
            string name
            string number PK
        }
        Teams {
            string name PK
        }
    
```

e.g. number (Players) and name (Teams) is a key for **Players**.

55

DESIGN TECHNIQUES

1. Avoid Redundancy.
2. Limit the use of weak entity sets.
3. Don't use an entity set when an attribute will do.

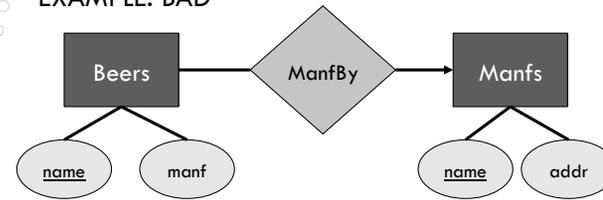
56

1. AVOIDING REDUNDANCY

- Redundancy is saying the same thing in two (or more) different ways.
 - or superfluity of information."
- Redundancy wastes space and (more importantly) encourages inconsistency.
 - Multiple representations of the same fact become inconsistent if we modify one and forget/do not modify its counterpart.

57

EXAMPLE: BAD

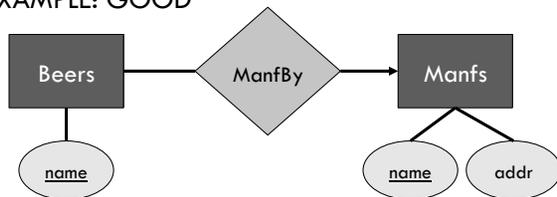


Note:

This design states the manufacturer of a beer twice: as an attribute and as a related entity set.

58

EXAMPLE: GOOD



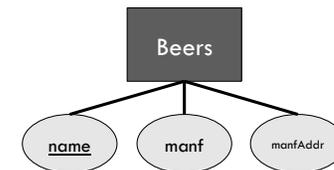
Note:

This design gives the address of each manufacturer exactly one.

59

EXAMPLE: BAD

Can somebody tell me why?



Note:

This design is repetitive! The manufacturer's address will be repeated for each beer. Also, in the event there are temporarily no beers for a manufacturer the value is lost.

60

ENTITY SETS VS. ATTRIBUTES

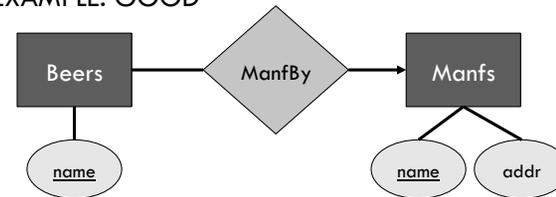
- An entity set should satisfy (at least) one of the following:
 - It is more than the name of something. i.e. it has at least one non-key attribute.

OR

- It is the "many" in a many-one or many-many relationship.
- Depends on the application requirements:
 - If we have several addresses per employee, `address` must be an entity (since attributes cannot be set-valued).
 - If the structure (city, street, etc...) is important, e.g., we want to retrieve employees in a given city, `address` must be modeled as an entity (since attribute values are atomic).

61

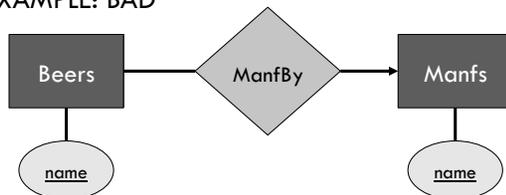
EXAMPLE: GOOD



- **Manfs** deserves to be an entity set because of the non-key attribute **addr**.
- **Beers** deserves to be an entity set because it is the "many" of the many-one relationship **ManfBy**.

62

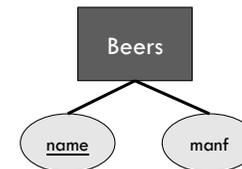
EXAMPLE: BAD



- Since the manufacturer is nothing but a name, and it is not at the "many" end of any relationship, it need not be an entity set.

63

EXAMPLE: GOOD



- There is no need to make the manufacturer an entity set, because we record nothing about manufacturers beside their name.

64

2. LIMIT THE USE OF WEAK ENTITY SETS

- Novice database designers often doubt that anything could be a key by itself.
 - They make all entity sets weak, supported by all other entity sets to which they are linked.
- In reality, we usually create unique ID's for entity sets.
 - Examples include: Social Insurance Numbers, automobile's VINs, etc...

65

WHEN DO WE NEED WEAK ENTITY SETS?

- The usual reason is that there is no 'global authority' capable of creating unique IDs.

e.g. it is unlikely that there could be an agreement to assign an unique player number across all football teams in the world.

66

FROM E/R DIAGRAMS TO RELATIONS

- Entity Set →
 - →
- →
 -
 -

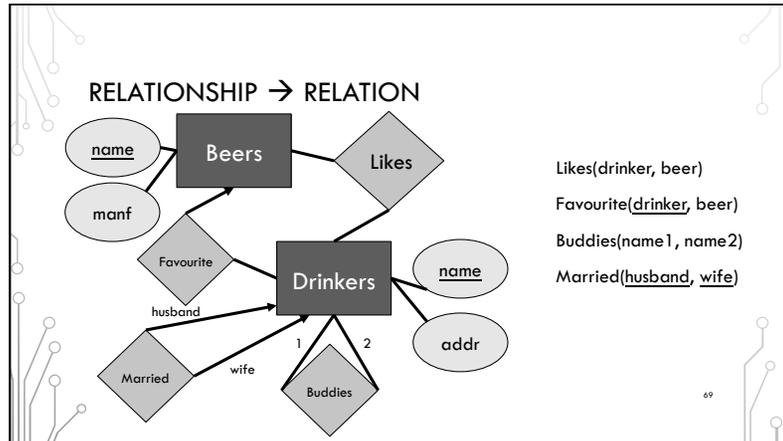
67

ENTITY SET → RELATION

Relation: Beers(name, manf)

```
graph TD; Beers[Beers] --- name((name)); Beers --- manf((manf));
```

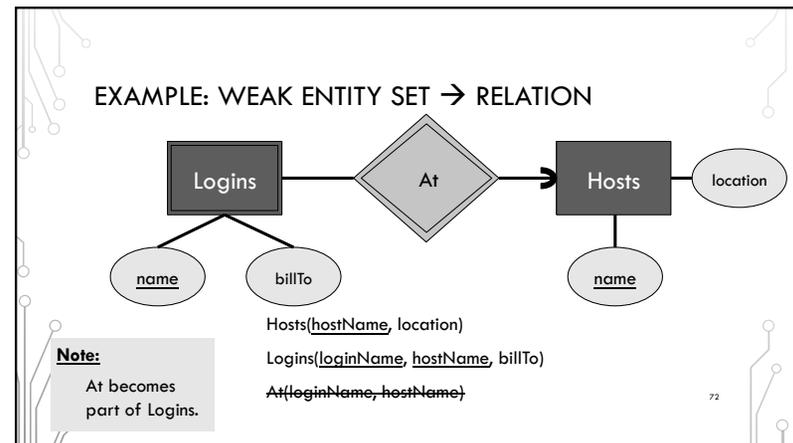
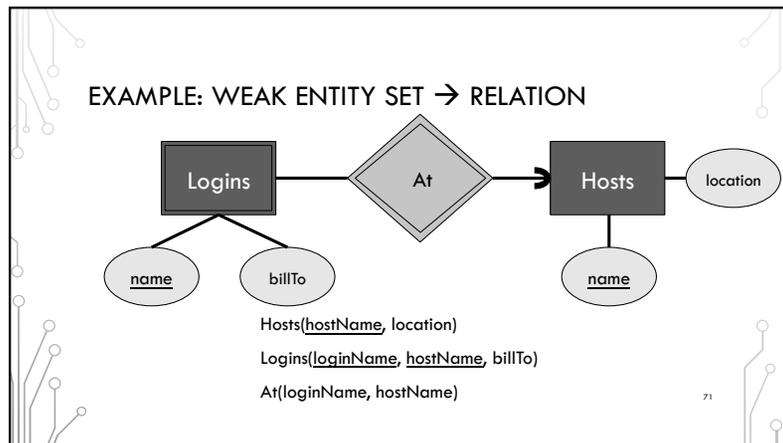
68



HANDLING WEAK ENTITY SETS

- Relation for a weak entity set must include attributes for its complete key (including those belonging to other entity sets), as well as its own, non-key attributes.
- A supporting relationship is redundant and yields no relation (unless it has attributes).

70



ENTITY VS. RELATIONSHIP

- ER diagram is OK if a manager gets a separate discretionary budget for each Departments.
- What if a manager gets a discretionary budget that covers all managed Departments?

- ❖ Redundancy:
 - ❖ dbudget stored for each dept managed by manager.
- ❖ Misleading:
 - ❖ Suggest dbudget associated with department manager combination.

73