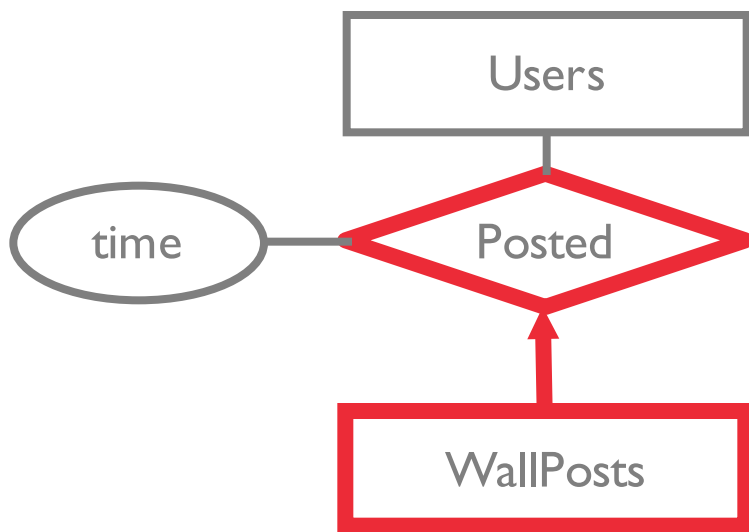


Weak Entity → Relation

Weak entity set and identifying relationship set are translated into a single table.

When owner entity is deleted, all owned weak entities also deleted.

pid in this example is a partial key: it alone does not identify a given wall post! For example, if each user's first wall post is given pid 1, then pid does not uniquely identify a wall post.



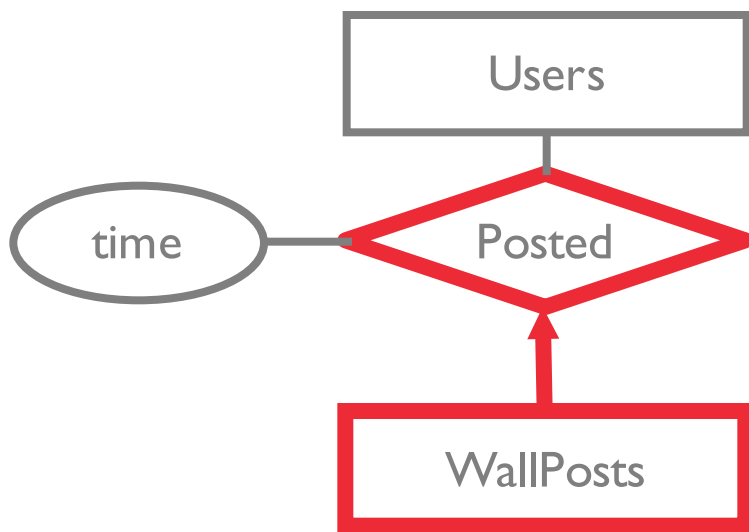
```
CREATE TABLE Wall_Posted(  
  pid int,  
  post_text text,  
  posted_at DATE,  
  uid int NOT NULL,  
  PRIMARY KEY (pid, uid),  
  FOREIGN KEY (uid) REFERENCES Users  
    ON DELETE CASCADE  
)
```

Weak Entity → Relation

Weak entity set and identifying relationship set are translated into a single table.

When owner entity is deleted, all owned weak entities also deleted.

In practice, `pid` in a `CREATE TABLE` statement is unique, so the primary key would simply be `pid`.



```
CREATE TABLE Wall_Posted(  
  pid int,  
  post_text text,  
  posted_at DATE,  
  uid int NOT NULL,  
  PRIMARY KEY (pid),  
  FOREIGN KEY (uid) REFERENCES Users  
  ON DELETE CASCADE  
)
```

ISA Hierarchies

Option 1: Keep base relation

Instructors & Students recorded in Users

Extra info in Instructors or Students relation

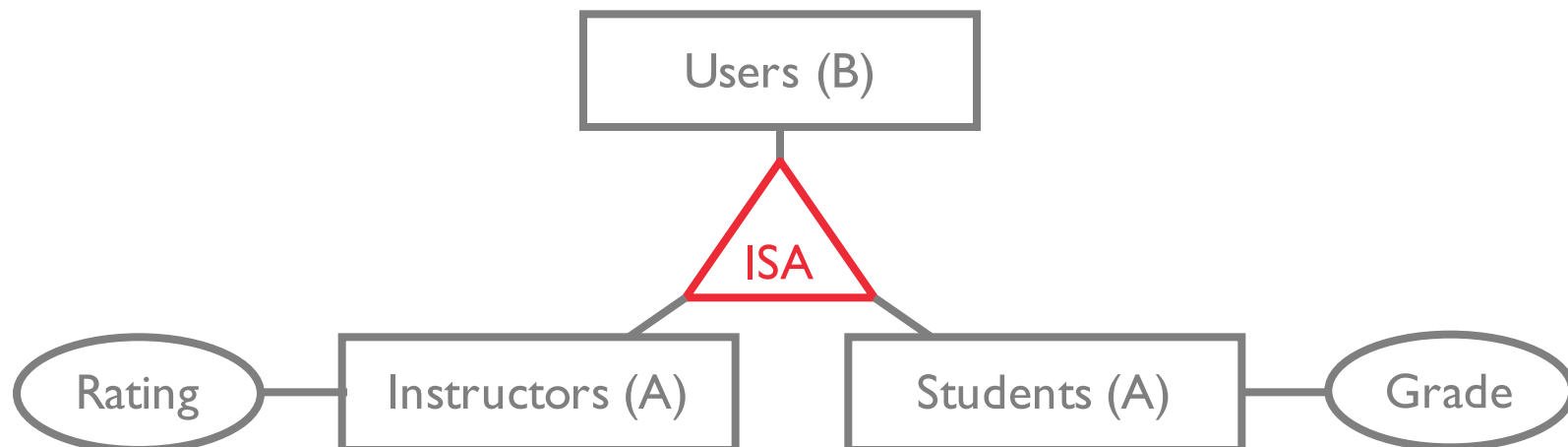
JOIN between child and base relations for all attributes

Option 2: Only keep child relations

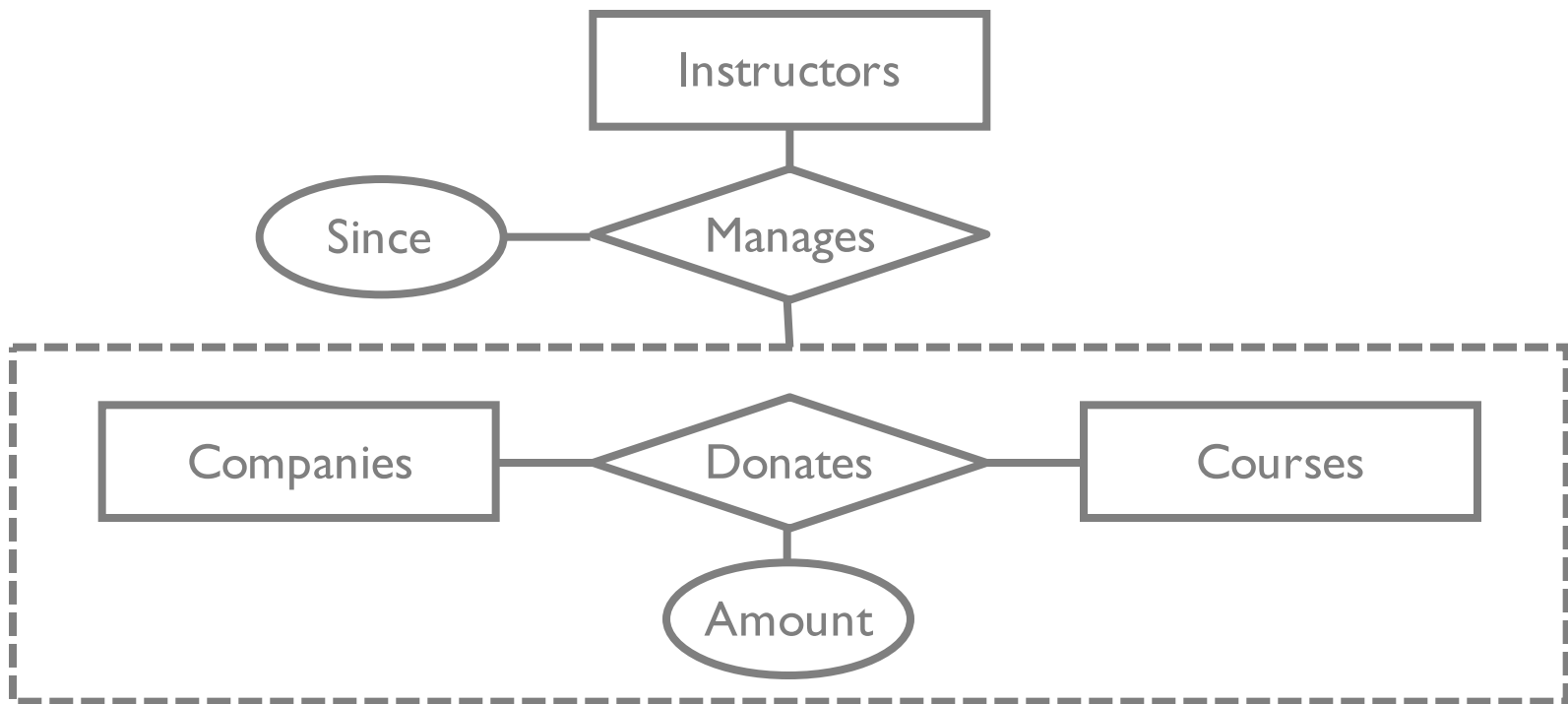
Instructors copies attributes from Users

Instructors(uid, name, age, ..., rating)

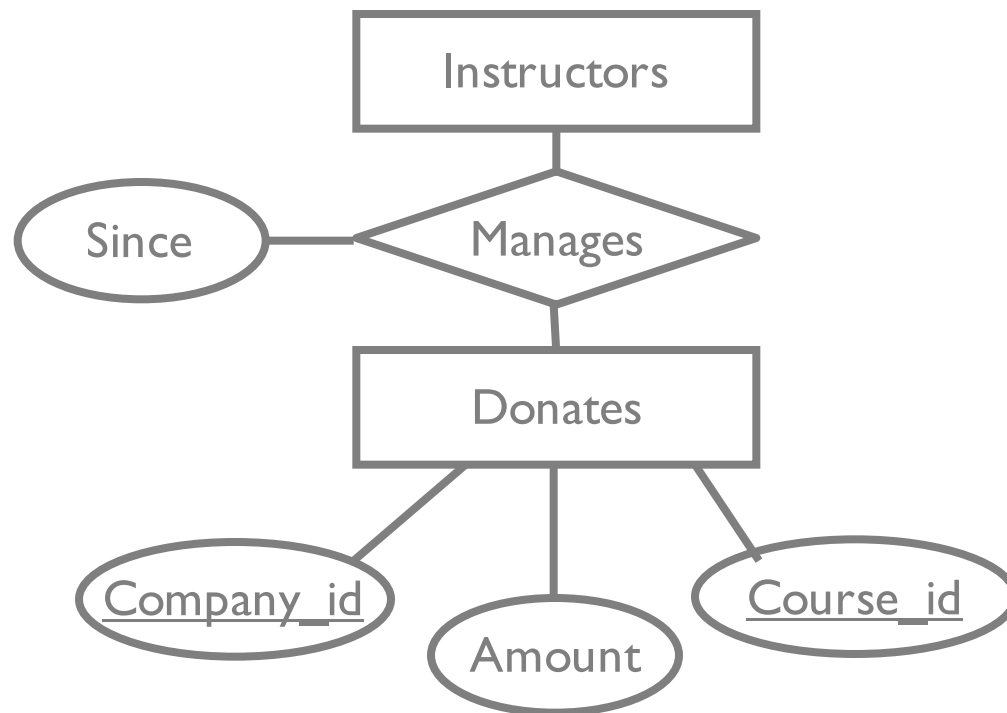
Only if covering
constraint = yes



Aggregation



Aggregation



in-class demo on instabase

<https://www.instabase.com/user/w4lll-nb/notebooks/ewu/w4lll-public/fs/Instabase%20Drive/Examples/lec5.ipynb>

note: the database doesn't support BEGIN transactions, so the cyclic references example doesn't run.

REVIEW OF ER AND RELATIONAL

ER Overview

Ternary relationships

Relationships constraints

- At most one

- At least one

- Exactly one

- Weak entities

Aggregation

Is-A

Relational Overview

Relations \sim Sets

Schema = structure

Instance = the data

Relational Review

Relations \sim Sets

Schema = structure

Instance = the data

"every relation is guaranteed to have a key" ?

Integrity Constraints

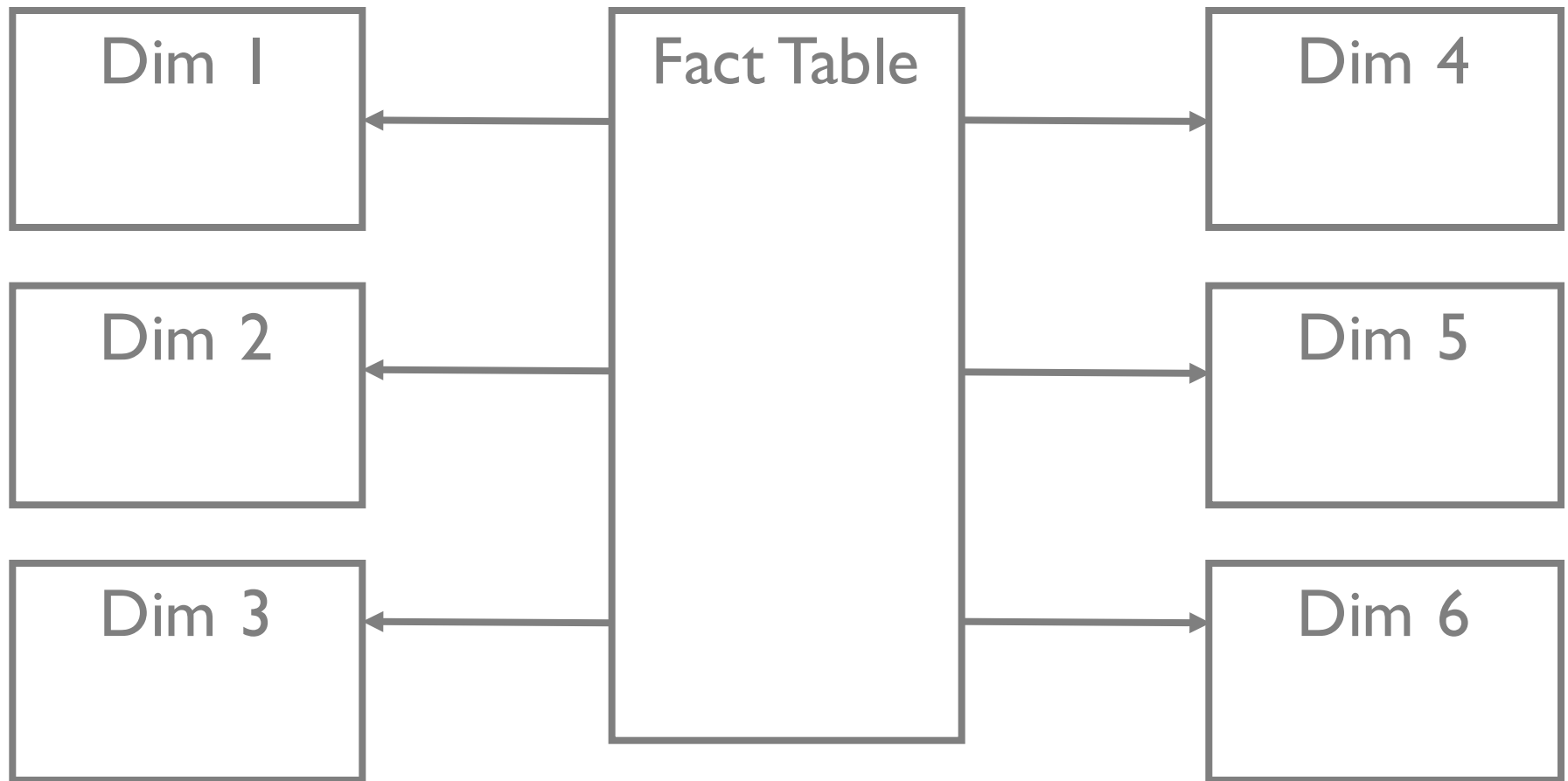
Candidate and primary keys

NOT NULL

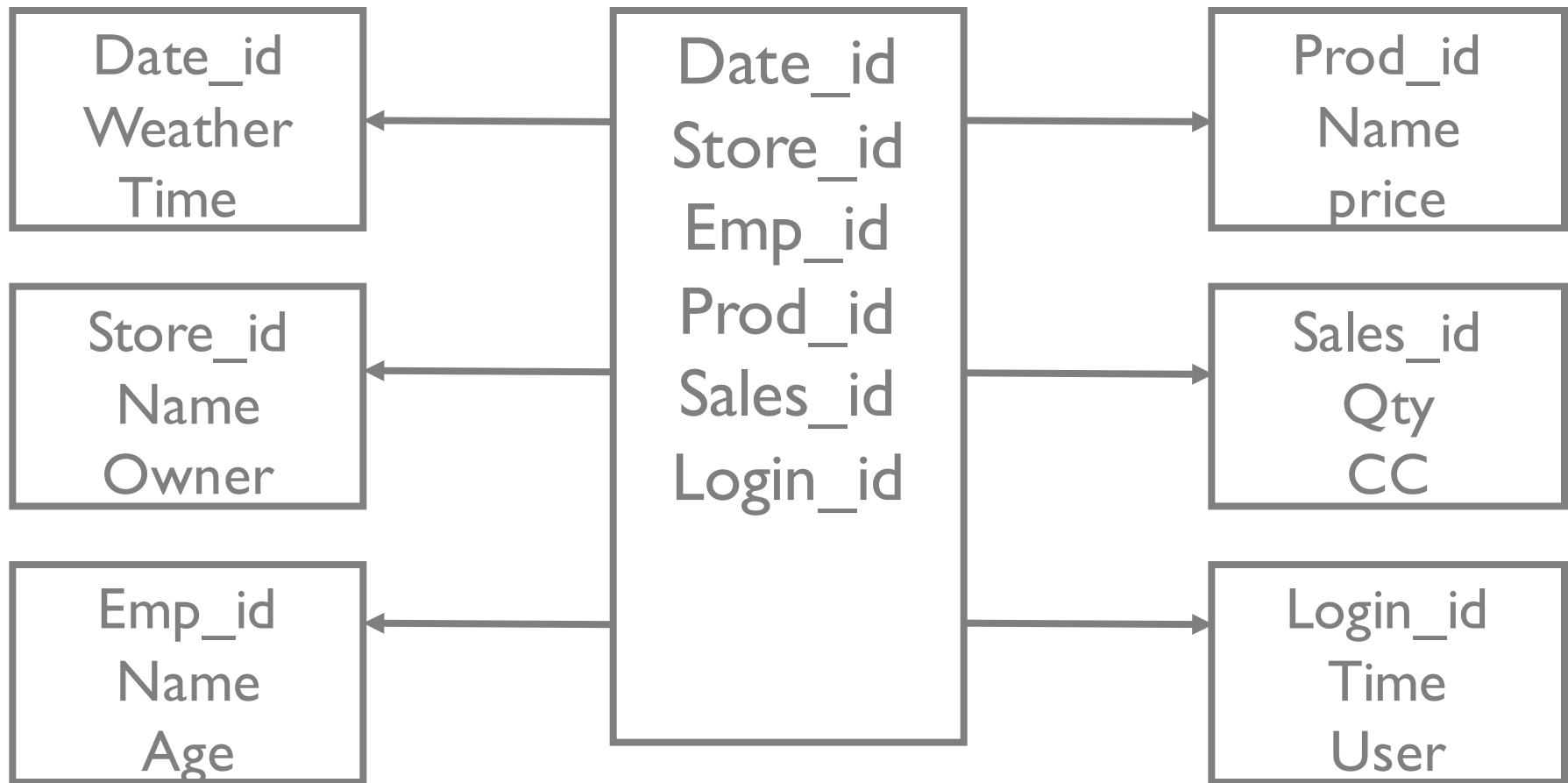
Referential integrity

How are foreign keys managed by the DBMS?

Star Schema



Star Schema



So What Happened?

1970 heated debates about CODASYL vs Relational
Network arguments

- low level languages more efficient (performance)

- relational queries would never be fast (performance)

Relational arguments

- data independence

- high level *simpler* languages

Market spoke.

Other models beyond relational!

Summary

Better than IMS/CODASYL

- allows us to talk about constraints!

- allows us to talk at a logical level

- declarative queries better than navigational programs

Everything is a relation (table)

DBA specifies ICs based on app, DBMS enforces

- Primary and Foreign Keys most used

- Types == Domain constraints

SQL