# Gagnasafnsfræði

# Einindavenslalíkanið (ER)

Hallgrímur H. Gunnarsson

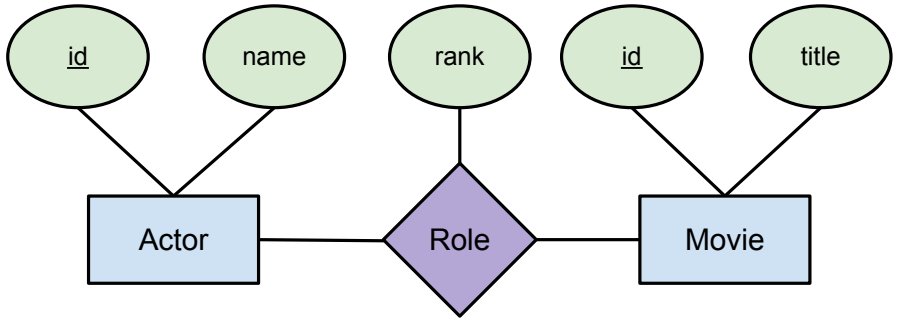
# Database design

1. Requirements analysis
   - user needs, what must the database capture?
2. Conceptual design
   - high level description (often done w/ER model)
3. Logical design
   - translate ER into DBMS data model
   - typically relational model as implemented by SQL
4. Schema refinement
   - consistency, constraints, normalization
5. Physical design
   - indexes, disk layout
6. Security design
   - who access what and how

# Entity-relationship model

Visual data model (diagram based)
- Top-down design
- Quickly sketch a database design
- Easier to "see" big picture

Basic concept: *entities* ("things") and their *relationships* to other entities, along with *attributes* describing them.



# Conceptual design using ER

What are the *entities* and their *relationships* ?

What info (attributes) about entities and relationships should be in the database?

What *integrity constraints* (business rules) hold?

ER diagram is a representation of a "schema"

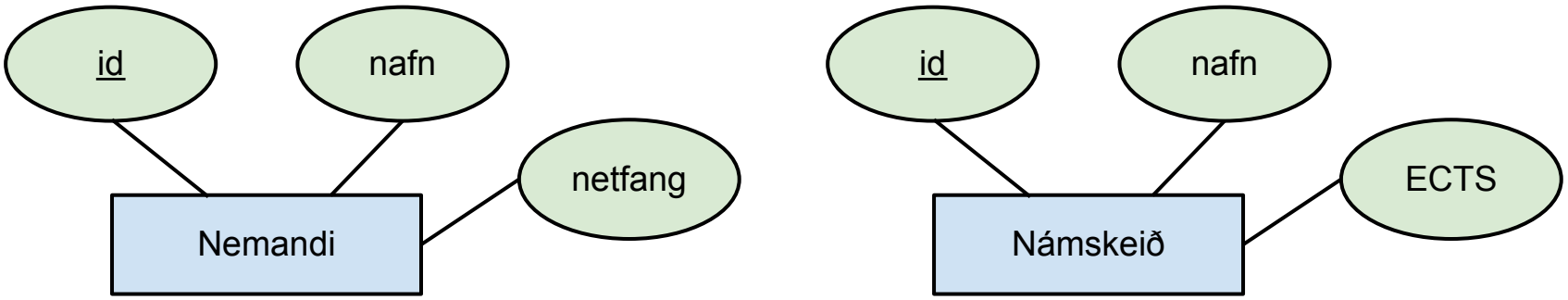
ER diagram can be mapped into a relational schema

# Entities

*Entity*: a "real-world" object described by a set of *attribute values*, e.g. a book. Each attribute has a *domain* of possible values.

*Entity Set*: a collection of similar entities, e.g. all books. All entities in a given entity set have the same attributes.

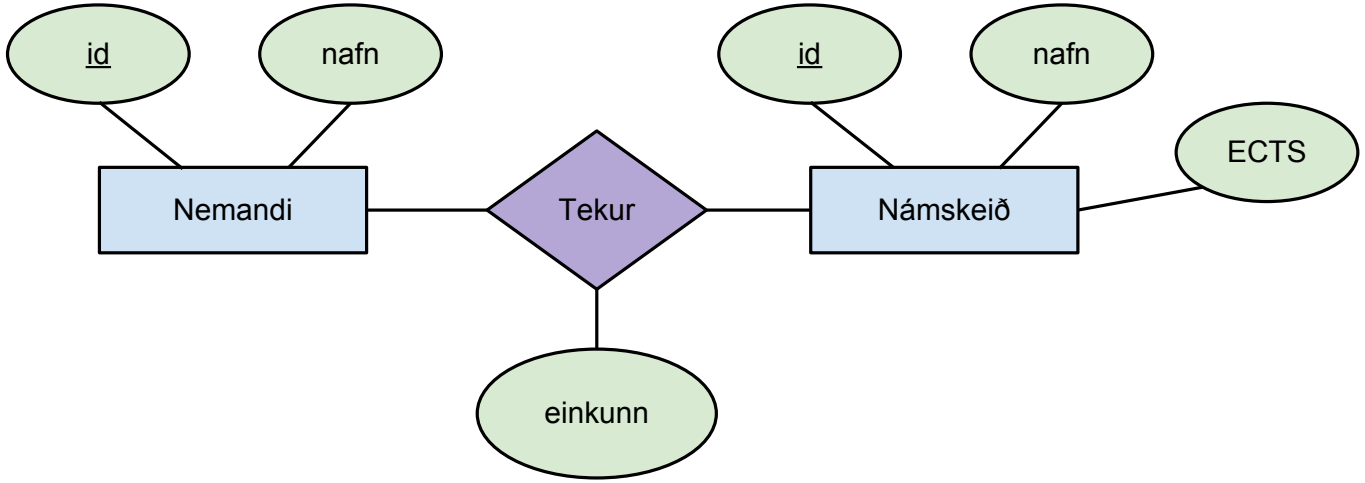
Every entity (unless it is a weak entity) has a minimal set of uniquely identifying attributes, called its primary key



# Relationships

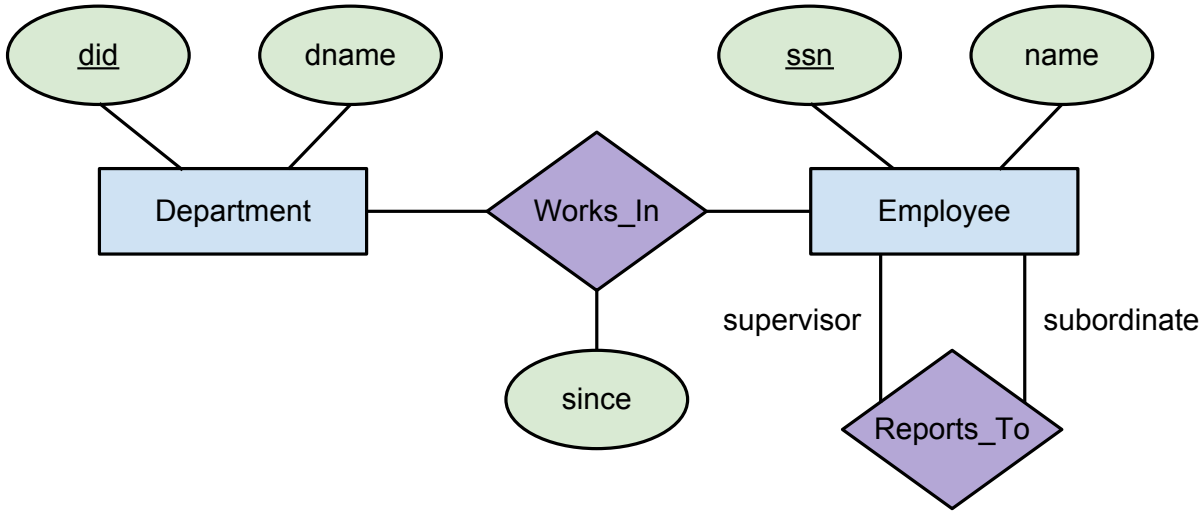
*Relationship*: an association among two or more entities, e.g. student X takes class Y. Relationships can have their own attributes.

*Relationship Set*: set of $n$-tuples. Each tuple denotes a relationship involving $n$ entities, $e_1 \ldots e_n$, where $e_i \in E_i$



# Roles in relationships

The same entity can participate in different relationship sets or in different "roles" in the same relationship set.

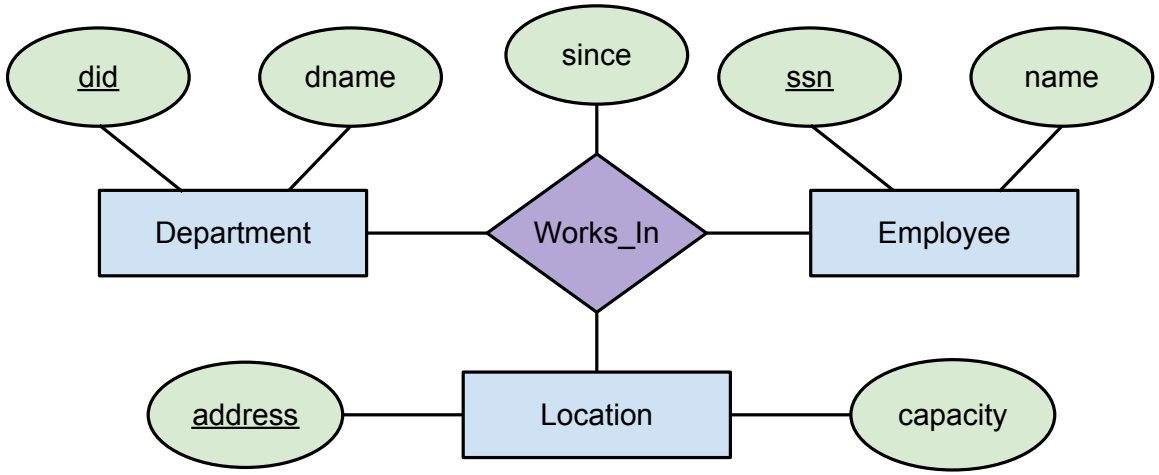


# Multiway relationships

Binary relationships not always enough.

Example: International company with multiple locations.

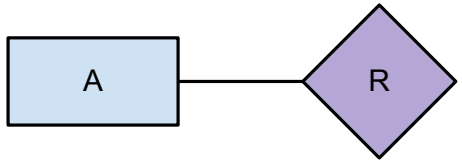
Employee X works in department Y at location Z.



# Constraints

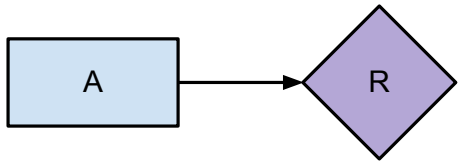
A regular line from an entity set to a relation set indicates no constraints.

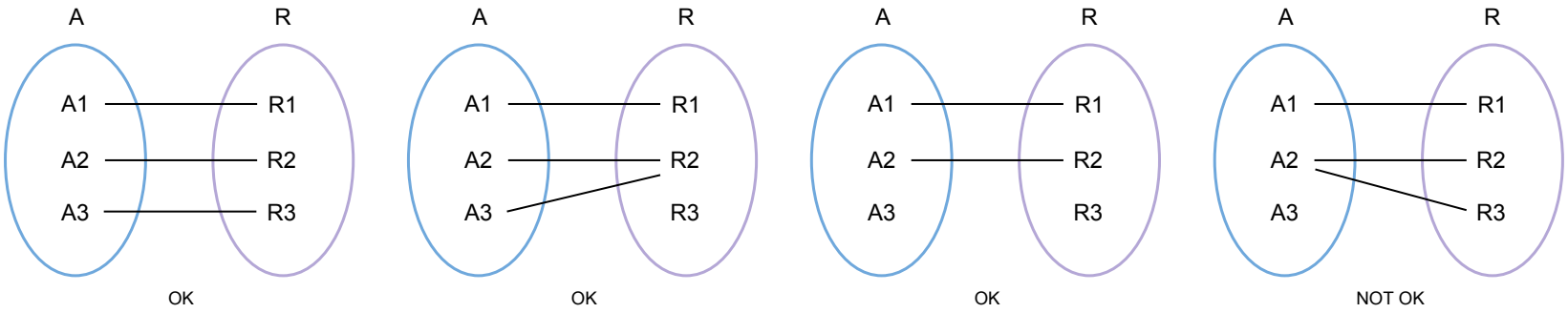
Each entity may be participate in any number of relationships.



# Key constraint (arrow)

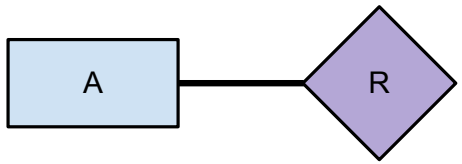
An arrow from an entity set to a relationship set indicates a key constraint. Each entity of the entity set can participate in *at most one* relationship in the relationship set.

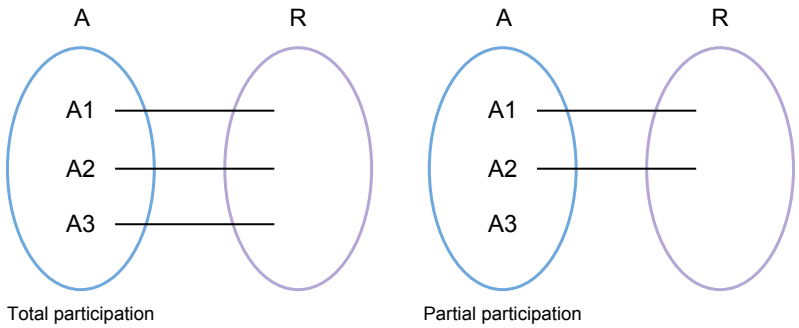


# Participation constraint (thick line)

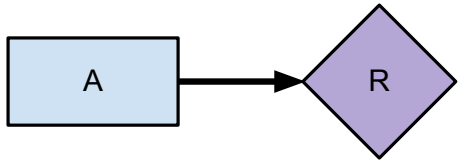
A thick line indicates a participation constraint. All entities in the entity set must participate in *at least one* relationship in the relationship set.

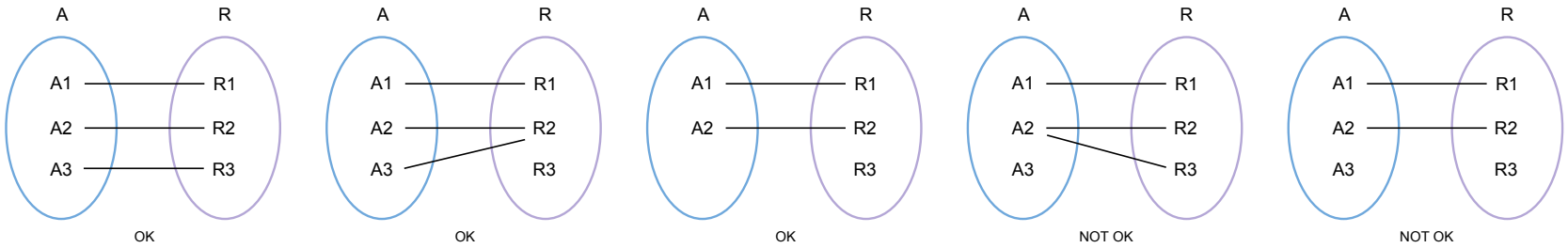


# Key + Participation constraint

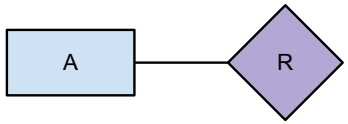
A thick line with an arrow indicates both a key and patricipation constraint. Each entity in the entity set is involved in *exactly one* relationship in the relationship set.



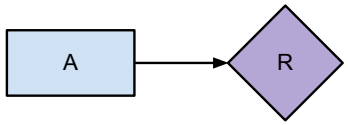



# Summary

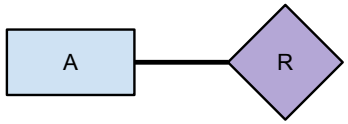
Entity may appear in any number of relations in R, including none:



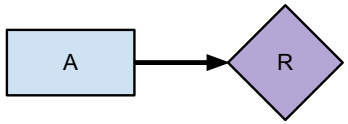

An entity in A may appear in at most one relation in R (key constraint):




Every entity in A must be a member of at least one relation in R (total constraint):




Every entity in A must be a member of exactly one relation in R (both):



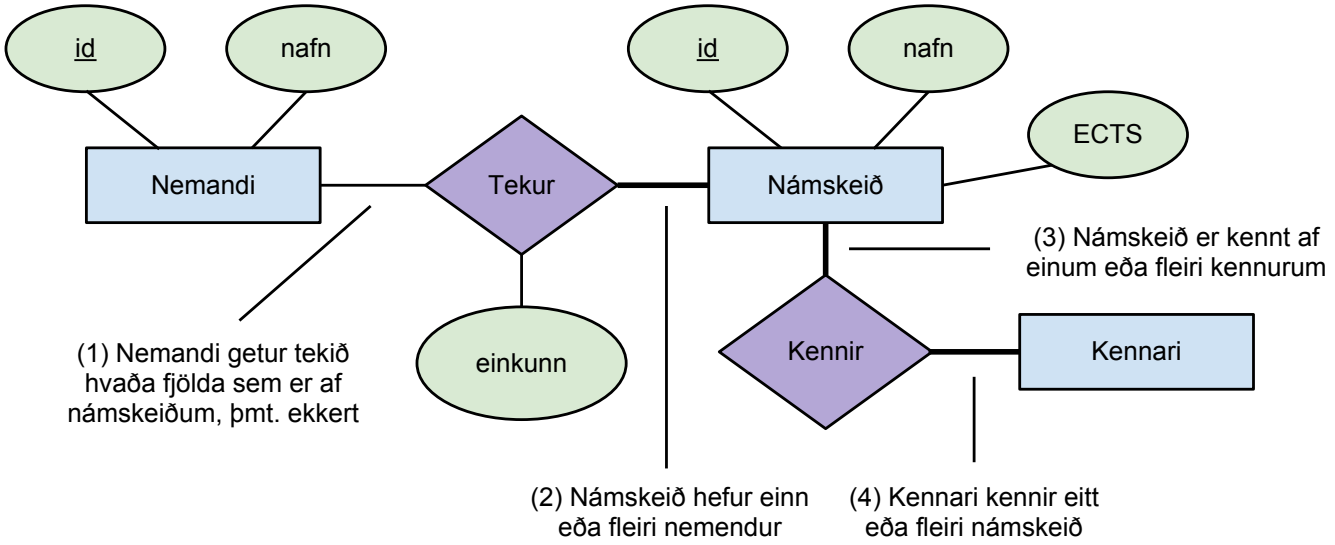
# ER example 1

Q1: What happens if we add a key constraint to nr. 1?

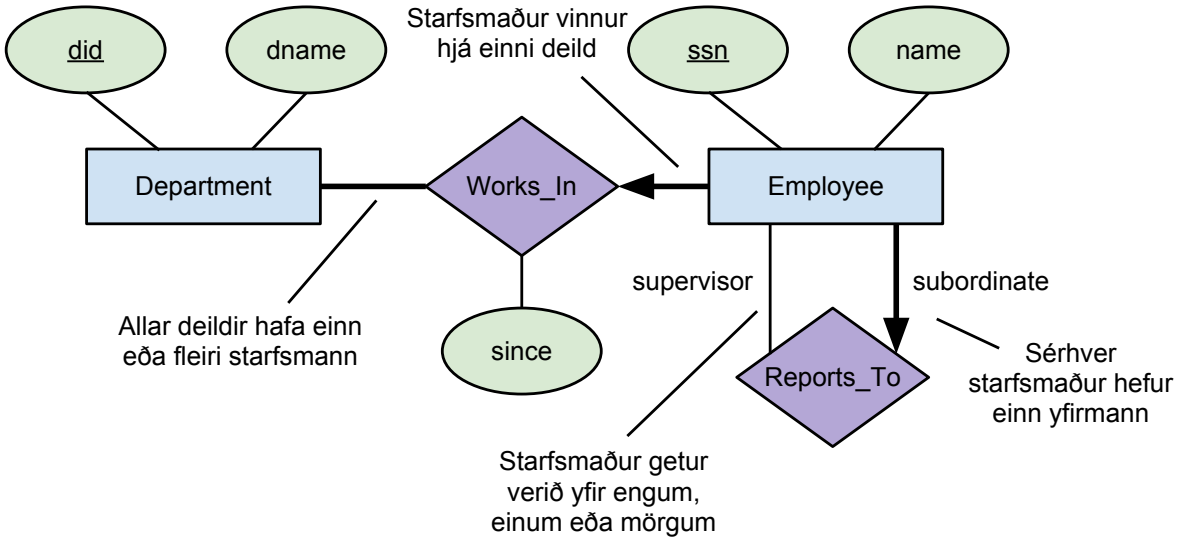
Q2: What happens if we add a key constraint to nr. 3?

Q3: What about a max constraint on the number of courses a student may take?

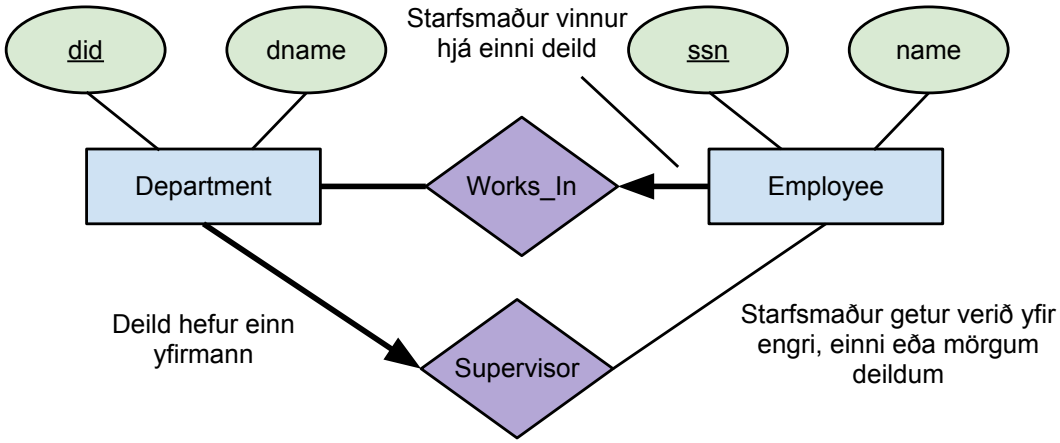
Q4: or max ECTS?



# ER example 2


did
dname
Starfsmaður vinnur
hjá einni deild
ssn
name
Department
Works_In
Employee
Allar deildir hafa einn
eða fleiri starfsmann
since
supervisor
subordinate
Reports_To
Sérhver
starfsmaður hefur
einn yfirmann
Starfsmaður getur
verið yfir engum,
einum eða mörgum

# ER example 3


did
dname
Starfsmaður vinnur hjá einni deild
ssn
name
Department
Works_In
Employee
Deild hefur einn yfirmann
Supervisor
Starfsmaður getur verið yfir engri, einni eða mörgum deildum

# ER example 4

Q1: Where does price go?

Q2: If price is an attribute of Book. What if the price changes?