Where We Are

- Now you know about:
  - What Database Management Systems are.
  - What Entity Relationship Models are.
  - Represent the Entity Relationship models using Entity Relationship Diagrams by Crow's Foot Notation.
- Entity Relationship Models are conceptual models.
- We need to convert ERDs to Relational Models so we are able to implement them as physical Databases.

Learning Objectives

- Finishing this module, you will be able to:
  - Understand and explain what are Relational Models
  - Explain the properties of Relational Models
  - Explain and select appropriate keys for Relational Models
  - Represent Relational Models with a set of Relational Schemas
  - Convert ERADs to Relational Models.

Relational Models

- Why are Relational Models?
  - Relational Model organizes data in two-dimensional tables: columns and rows.
  - Relational Model includes: Relations, Tuples, Attributes, keys and foreign keys.
  - Relational Models are represented by a set of Relational Schemas.

Relation Examples

Stores

StoreID	Street	City	Zip
#1506	1200 W Dillon Rd	Louisville	80027
#1546	1600 29th Street	Boulder	80301
#1524	1271 Sheridan Blvd	Broomfield	80020
#1517	7125 W 88th Ave	Westminster	80021
#1548	16420 Washington Street	Thornton	80023
#1503	10003 Grant Street	Thornton	80229
#1502	5215 Wadsworth Blvd	Arvada	8002

Cardinality

Degree

Relation Examples

Employees

EmpID	FirstName	LastName	DoB	Position	Departme	StoreID
#20399	John	Ford	1998/2/12	Manager	HR	#1506
#30123	Anne	Brand	2001/3/12	Intern	Marketing	#1546
#12524	David	Biden	2000/2/20	Assistant	Sales	#1524
#14517	William	Potter	2001/9/12	Senior Manager	HR	#1506
#15214	Mary	Alexander	2001/9/12	Assistant	IT	#1524
#11032	Rose	Smith	1999/1/21	Intern	IT	#1503
#02012	Julie	Smith	1977/12/1	Senior Manager	IT	#1503
#78123	Angela	White	1967/4/4	Senior Manager	HR	#1546
#21342	John	Ford	1983/11/11	Manager	IT	#1546

Cardinality

Degree

Terminologies

- A relation is a table with columns and rows.
  - Attribute is a named column of a relation.
  - Domain is the set of allowable values for one or more attributes.
  - Tuple is a row of a relation.
  - Degree is the number of attributes in a relation.
  - Cardinality is the number of tuples in a relation.
- Relational Database is a collection of normalized relations with distinct relation names.

Relational Keys

- Superkey
  - An attribute, or a set of attributes, that uniquely identifies a tuple within a relation.
- Candidate Key
  - Superkey (K) such that no proper subset is a superkey within the relation.
    - In each tuple of R, values of K uniquely identify that tuple (uniqueness).
    - No proper subset of K has the uniqueness property (irreducibility).

Relational Keys

- Primary Key
  - Candidate key selected to identify tuples uniquely within relation.
- Alternate Keys
  - Candidate keys that are not selected to be primary key.
- Foreign Key
  - Attribute, or set of attributes, within one relation that matches candidate key of some (possibly same) relation.

Relation Examples

Primary Key

StoreID	Street	City	Zip
#1506	1200 W Dillon Rd	Louisville	80027
#1546	1600 29th Street	Boulder	80301
#1524	1271 Sheridan Blvd	Broomfield	80020
#1517	7125 W 88th Ave	Westminster	80021
#1548	16420 Washington Street	Thornton	80023
#1503	10003 Grant Street	Thornton	80229
#1502	5215 Wadsworth Blvd	Arvada	8002

Foreign Key

EmpID	FirstName	LastName	DoB	Position	Department	StoreID
#20399	John	Ford	1998/2/12	Manager	HR	#1506
#30123	Anne	Brand	2001/3/12	Intern	Marketing	#1546
#12524	David	Biden	2000/2/20	Assistant	Sales	#1524
#14517	William	Potter	2001/9/12	Senior Manager	HR	#1506
#15214	Mary	Alexander	2001/9/12	Assistant	IT	#1524
#11032	Rose	Smith	1999/1/21	Intern	IT	#1503
#02012	Julie	Smith	1977/12/1	Senior Manager	IT	#1503
#78123	Angela	White	1967/4/4	Senior Manager	HR	#1546
#21342	John	Ford	1983/11/11	Manager	IT	#1546

Database Relations

- Relation schema
  - Defines a relation by a set of attributes (and their domain).
- Relational database schema
  - Set of relation schemas, each with a distinct name.
- General format:
  - Name(Attribute$_1$, Attribute$_2$, …, Attribute$_x$(fk), … , Attribute$_N$)
  - The attribute(s) with underline as key
  - The attribute(s) with (fk) as foreign key(s)

Relational Schema Examples

Stores

StoreID	Street	City	Zip
#1506	1200 W Dillon Rd	Louisville	80027
#1546	1600 29th Street	Boulder	80301
#1524	1271 Sheridan Blvd	Broomfield	80020
#1517	7125 W 88th Ave	Westminster	80021
#1548	16420 Washington Street	Thornton	80023
#1503	10003 Grant Street	Thornton	80229
#1502	5215 Wadsworth Blvd	Arvada	8002

- Stores(StoreID, Street, City, Zip)

Relational Schema Examples

Employees

EmpID	FirstName	LastName	DoB	Position	Departme	StoreID
#20399	John	Ford	1998/2/12	Manager	HR	#1506
#30123	Anne	Brand	2001/3/12	Intern	Marketing	#1546
#12524	David	Biden	2000/2/20	Assistant	Sales	#1524
#14517	William	Potter	2001/9/12	Senior Manager	HR	#1506
#15214	Mary	Alexander	2001/9/12	Assistant	IT	#1524
#11032	Rose	Smith	1999/1/21	Intern	IT	#1503
#02012	Julie	Smith	1977/12/1	Senior Manager	IT	#1503
#78123	Angela	White	1967/4/4	Senior Manager	HR	#1546
#21342	John	Ford	1983/11/11	Manager	IT	#1546

- Employees (EmpID, FirstName, LastName, DoB, Position, Department, StoreID(fk))

Properties of Relations

- Each tuple is distinct; there are no duplicate tuples.
- Order of attributes has no significance.
- Order of tuples has no significance, theoretically.
- Each cell of relation contains exactly one value.
- Each attribute has a distinct name.
- Values of an attribute are all from the same domain.
- Relation name is distinct from all other relation names in a Relational Model.

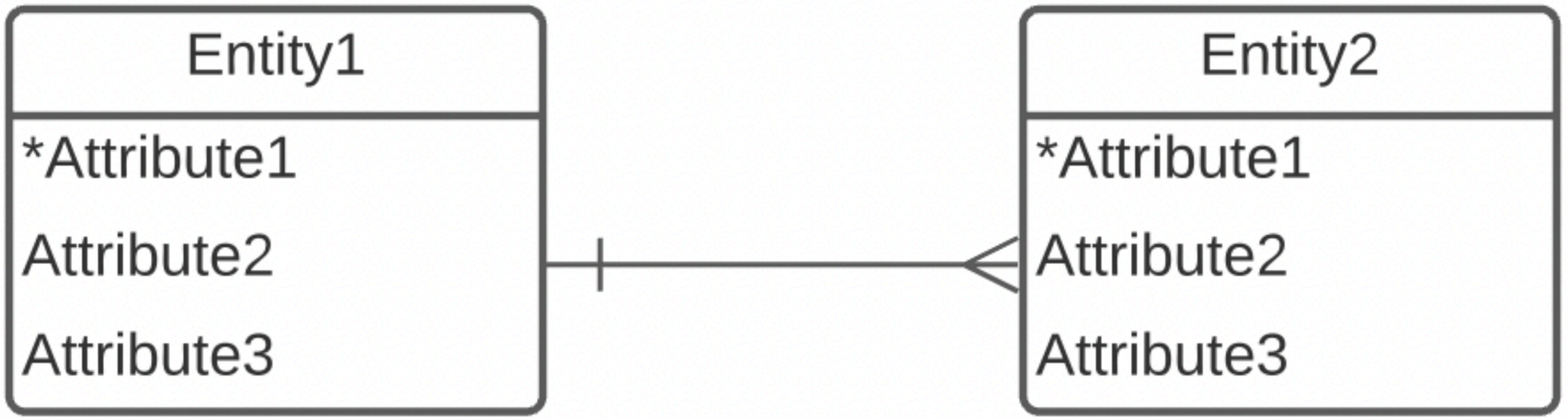
Practice

- Let's do more practice in Lab1.

Converting ERADs to RMs

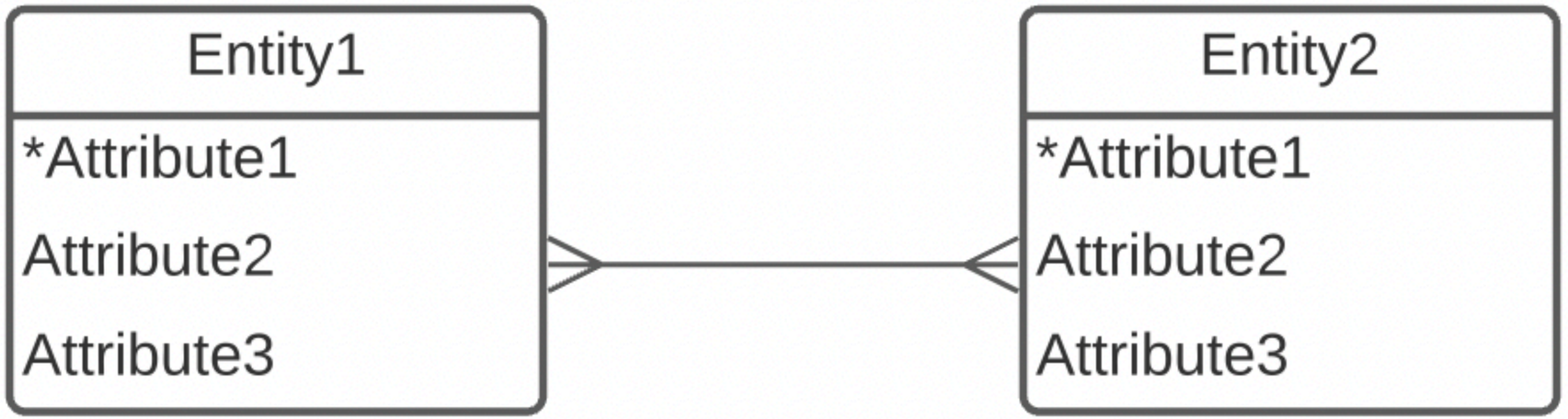
- Now we need to convert ERADs to Relational Models.
- For a majority of ER Models, entities and weak entities convert easily into relations.
- General steps:
  - Each entity will be converted directly to a relation.
  - The attributes of the entity become the attributes of the relation.
  - The Identifier of the Entity becomes a Key of the relation.
  - Relationships will be mapped as foreign keys.
    - Participation will be ignored at this step, only cardinality matters.

Binary, One to Many



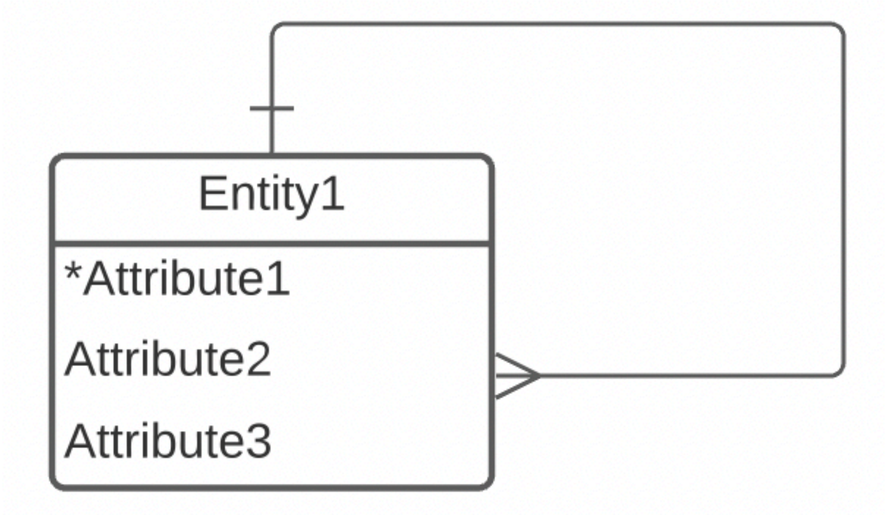
- Entity1(Attribute1, Attribute2, Attribute3)
- Entity2(Attribute1, Attribute2, Attribute3, Attribute1B(fk))

Binary, Many to Many



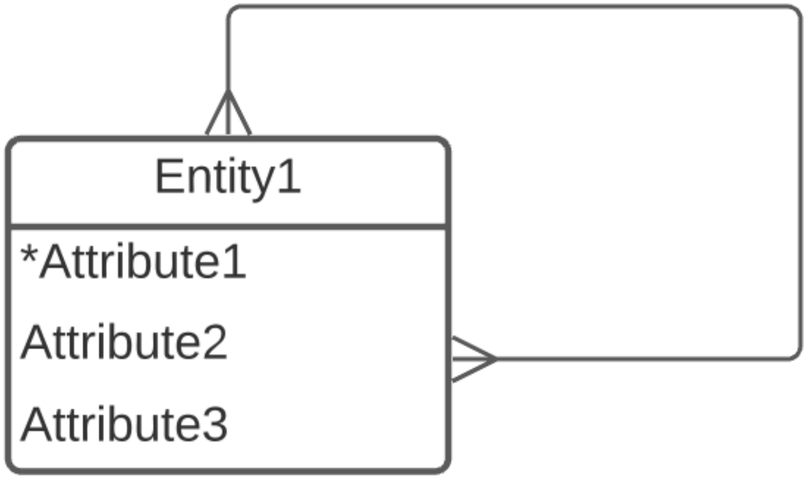
- Entity1(Attribute1, Attribute2, Attribute3)
- Entity2(Attribute1, Attribute2, Attribute3)
- Entity1_2(Attribute1A(fk), Attribute1B(fk))

Unary, One to Many



- Entity1(Attribute1, Attribute2, Attribute3, Attribute1B(fk))

Unary, Many to Many



- Entity1(Attribute1, Attribute2, Attribute3)
- Entity1_1(Attribute1A(fk), Attribute1B(fk))

One to One

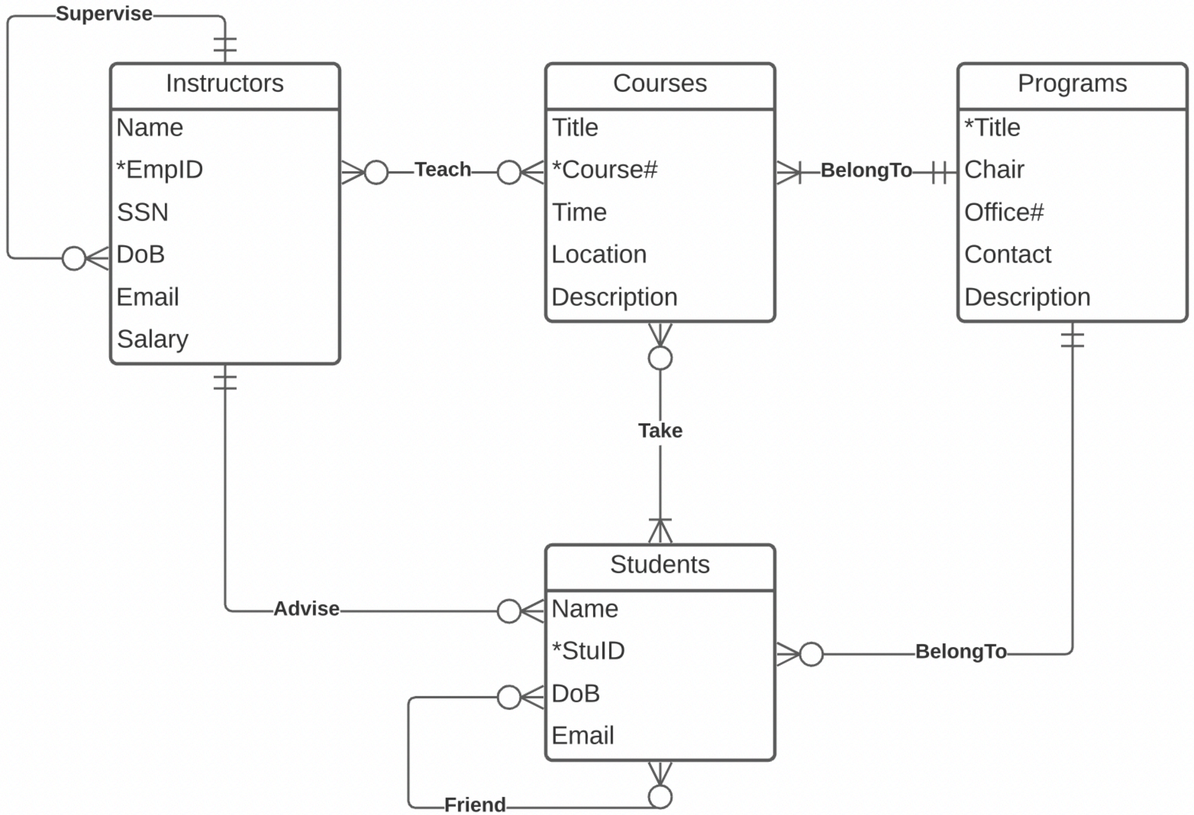
- Merge:
  - We can merge two entities as one, and find a primary key for the new entity.
- Not Merge:
  - One to One can be treated as a special case of One to Many.
  - If both sides are mandatory, we can choose either side as one, and choose the other side as many.
  - If only one side is mandatory, we choose this side as one, and the other as many.
  - If both sides are optional, we have to rethink about the primary key.

More Than Binary

- If the relationship has more than 2 entities evolved, then we need to find a way to separate them to some binary/unary relationships.
- We cannot handle these relationships.

Practice

- Let's convert the ERAD to a Relational Model



Instructors(Name, EmpID, SSN, DoB, Email, Salary, SupervisorID(fk))
Course(Title, Course#, Time, Location, Description, Title(fk))
Instructor_Course(EmpID(fk), Course#(fk))
Programs(Title, Chair, Office#, Contact, Description)
Students(Name, StuID, DoB, Email, Title(fk), EmpID(fk))
Students_Course(StuID(fk), Course#(fk))
Students_Students(StuID(fk), FriendID(fk))

Practice

- Let's do more practice in Lab2.

Assignment

- Let's do the assignment to assess your understanding of Relational Models, and converting ERDs to Relational Models..

Congratulations

- Now you finished Module 2!
- You should be comfortable to write Relational Schemas, and to convert Entity Relationship Diagrams to Relational Models.
- Next step is to normalize Relational Models to Third Normal Form for implementation.
- See you soon.