

Database Fundamentals – CS990

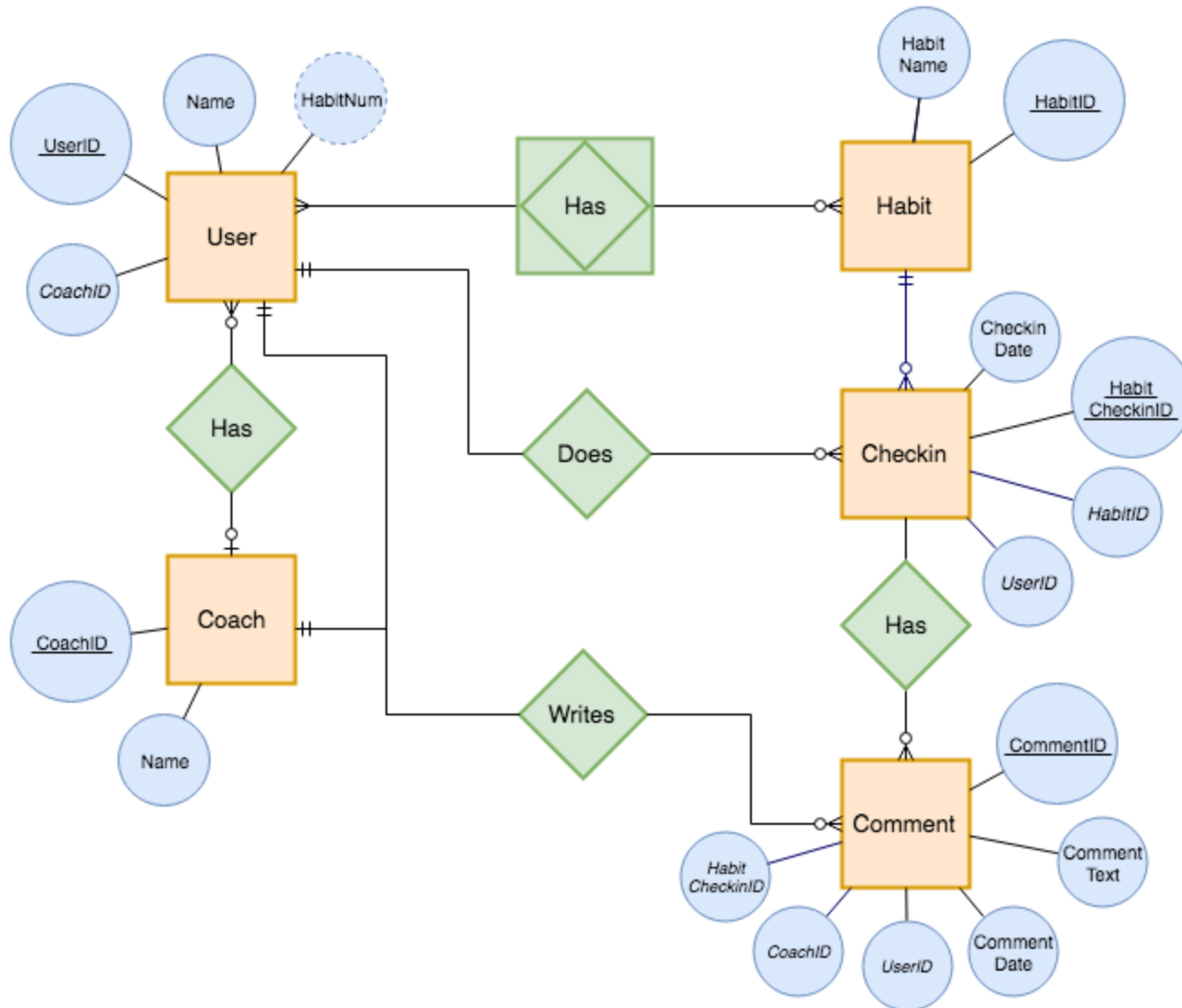
Database and Web Systems Development - CS952

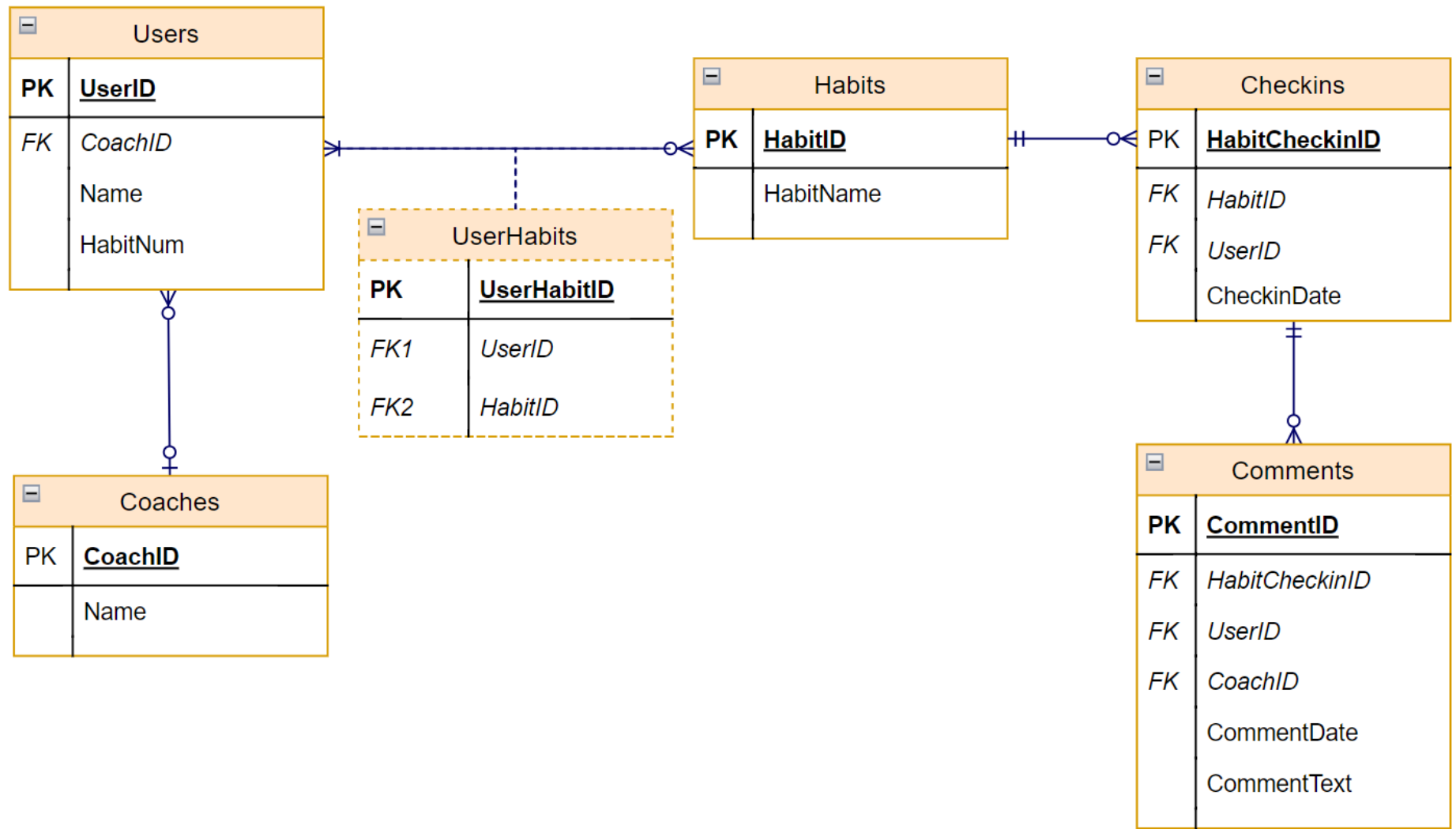
ER Modelling

Course Content

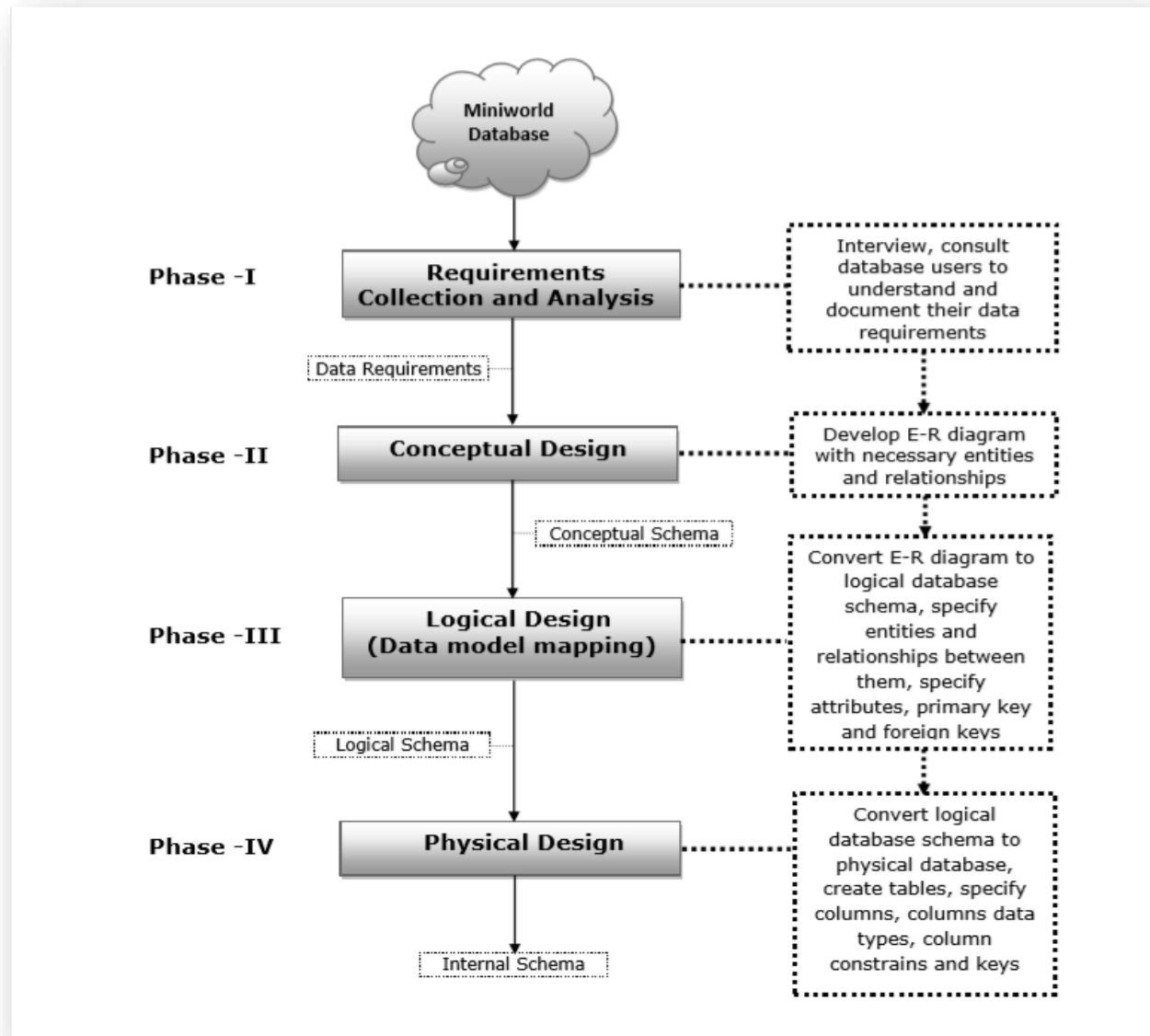
1. Introduction to Relational Databases (*Introduction + Relational Model*)
2. **Data Modelling** - (*Entity Relationship Modelling + The Enhanced Entity Relationship Model*)
3. Database Design and SQL - (*Logical modelling + Introduction to SQL*)
4. Further SQL - (*Advanced SQL queries + Creating tables with SQL*)
5. Normalisation - (*Normalisation to second normal form + Third normal form*)

Draw.io





Database Design Process Flowchart



Problems with ER Models

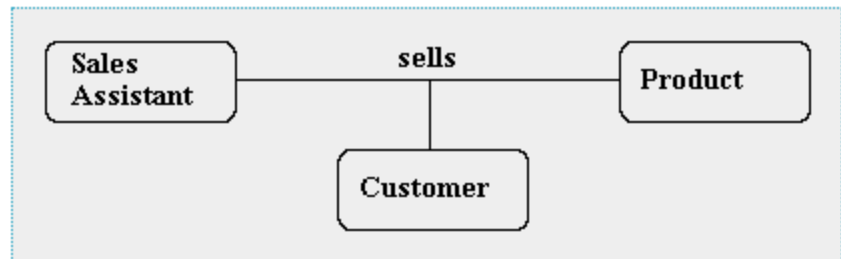
- There are problems that may arise when designing a conceptual data model. Some are known as connection traps.
- There are two main types of connection traps:
 - fan traps
 - chasm traps

Ternary relationship

Binary Relationships

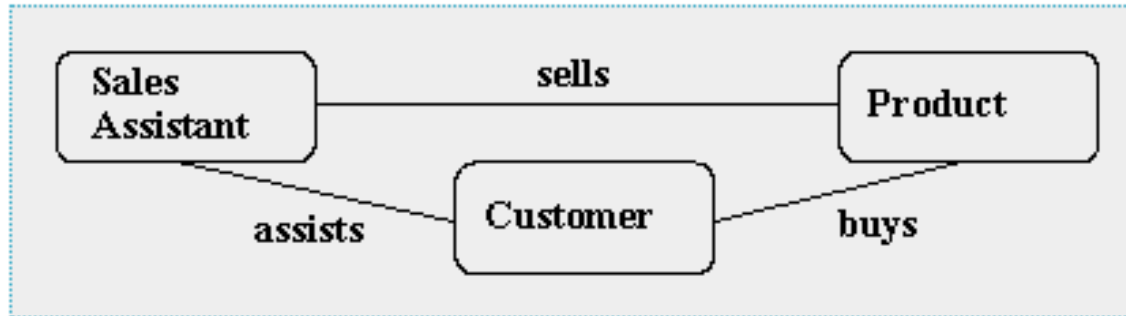


Ternary Relationships: If there are three entity types involved



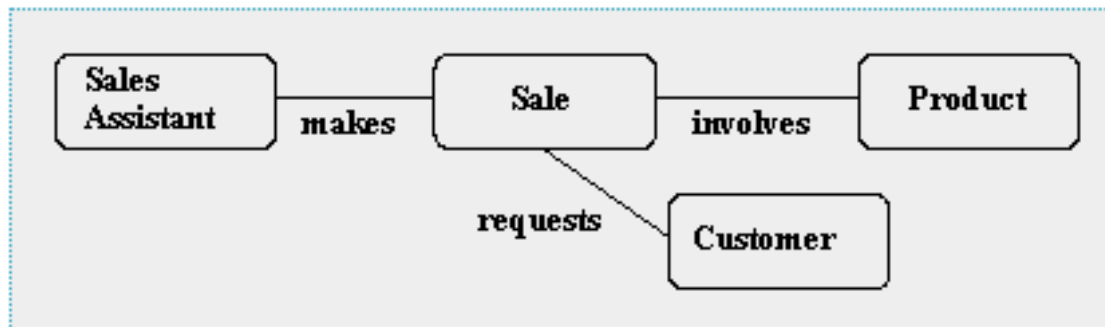
Replacing ternary relationships

Loss of some information - It is no longer clear which sales assistant sold a customer a particular product.



Replace: the ternary relationship with an entity type and a set of binary relationships.

So, a sales assistant can be linked to a specific customer and both of them to the sale of a particular product.



Fan traps

- A fan trap occurs when a model represents a relationship between entity types, but the pathway between certain entity occurrences is ambiguous. It occurs when 1:m relationships fan out from a single entity.
- A single site contains many departments and employs many staff. However, which staff work in a particular department?



Cont...

- A single site contains many departments and employs many staff. However, which staff work in a particular department?
- The fan trap is resolved by restructuring the original ER model to represent the correct association.



Example

Relationships considered so far are binary ie they involve two entities. It is possible for relationships to link three entities eg:

"A supplier may supply a range of parts to a number of projects. Each part is available from a number of suppliers. Each project uses a number of parts which are available from a number of suppliers.

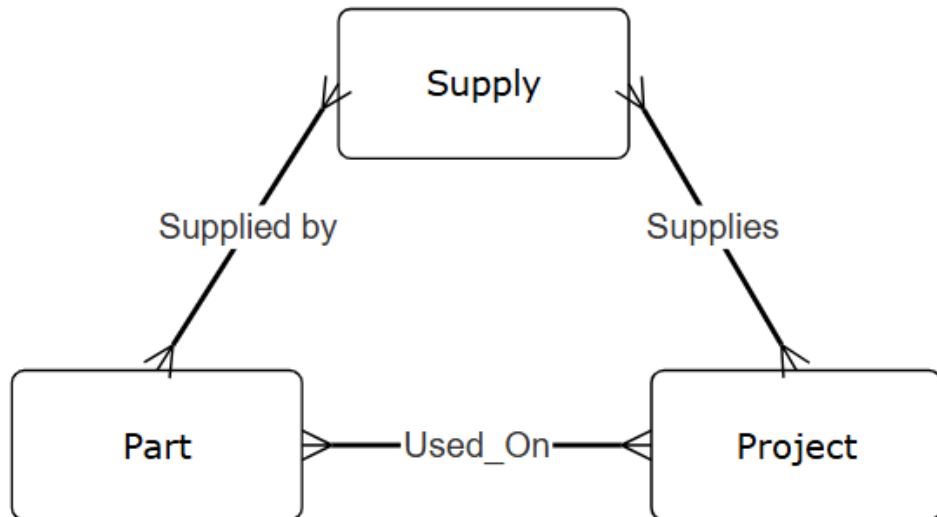
" Many-to-many relationships exist between:

Part and Project

Project and Supplier

Supplier and Project

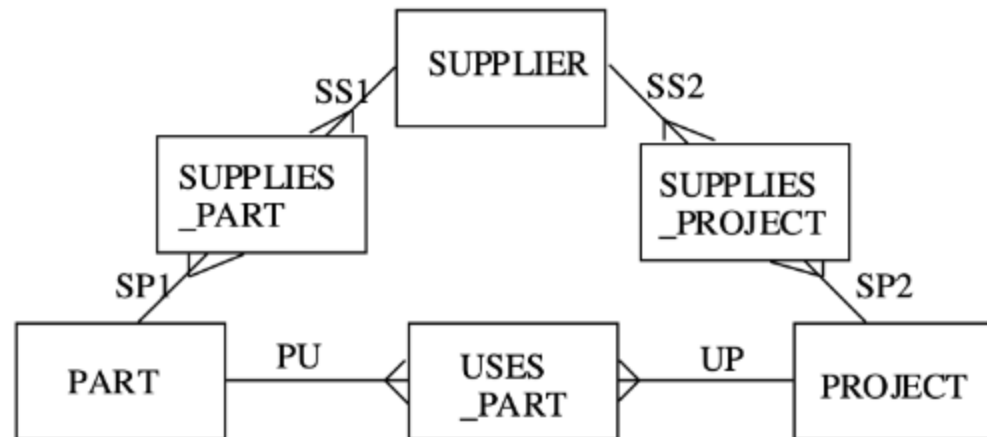
This could be represented by:



Decompose?

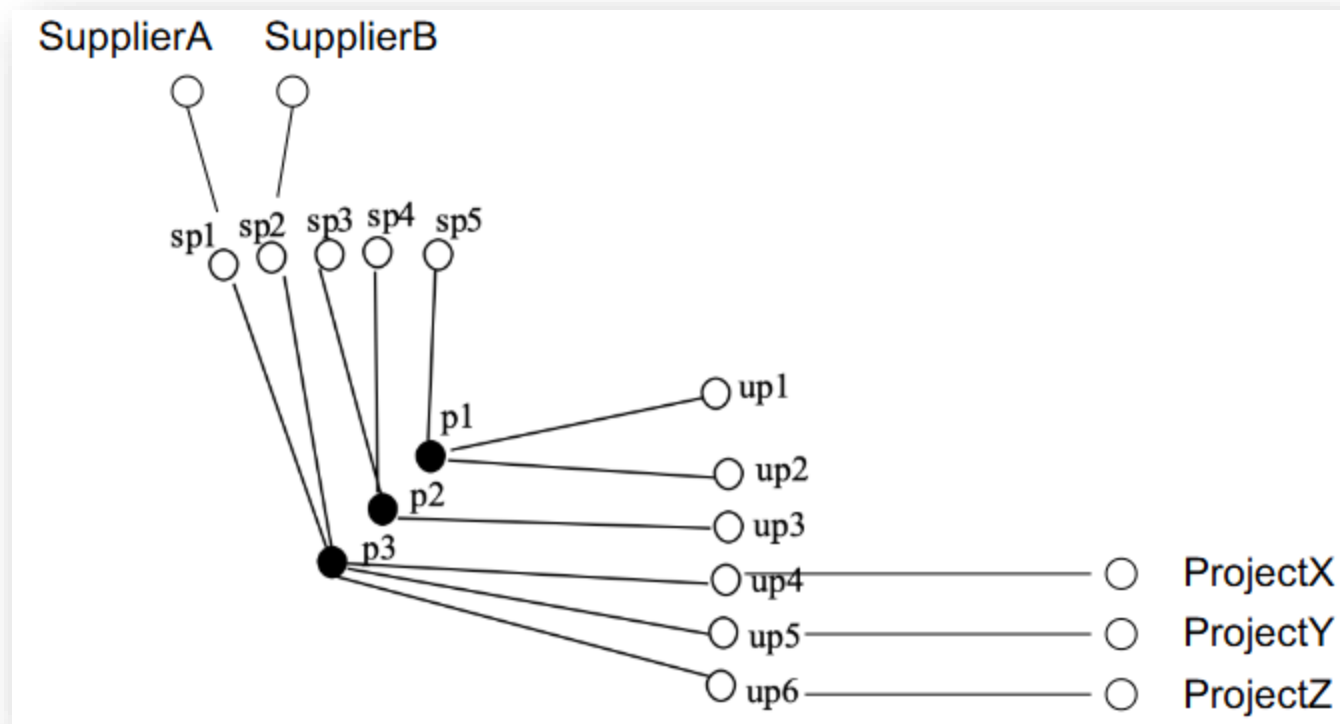
Decompose each one of these many to many relationships and put in a linker entity?

Decomposing the many-to-many relationships gives:



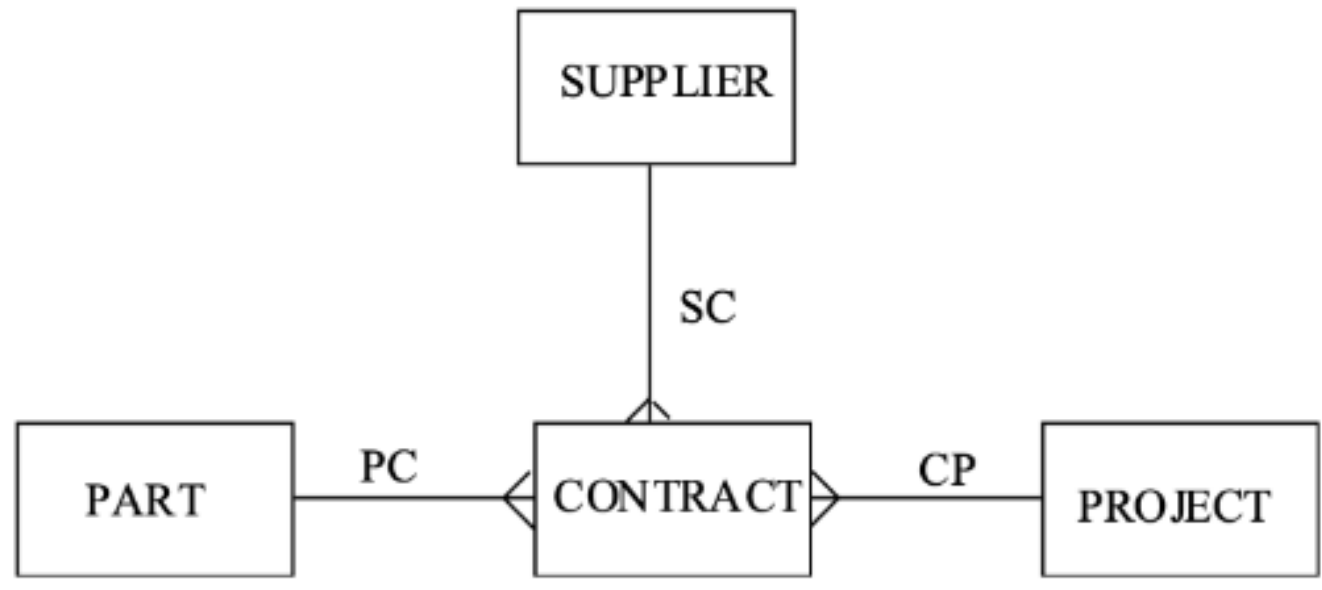
Occurrence Diagram

It looks as if there may be redundant relationships, but it is not possible to link PROJECT to SUPPLIER via PART because of fan traps. This is illustrated by an occurrence diagram for SUPPLIES_PART, PART and USES_PART:



Final ERD

A simpler entity relationship structure which represents the same information uses a single linker entity to represent the ternary relationship:



Chasm traps

- A chasm trap occurs when a model suggests the existence of a relationship between entity types, but the pathway does not exist between certain entity occurrences.
- It occurs where there is a relationship with partial participation, which forms part of the pathway between entities that are related.

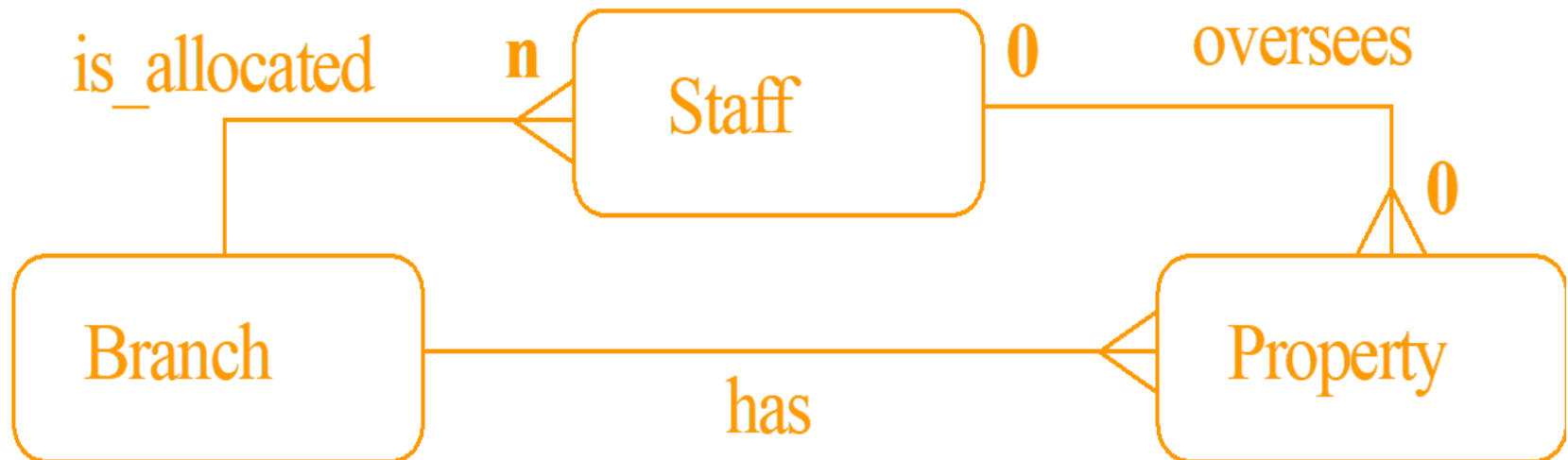


Cont...

- A single branch is allocated many staff who oversee the management of properties for rent. Not all staff oversee property and not all property is managed by a member of staff.
- What properties are available at a branch?
- The partial participation of Staff and Property in the oversees relation means that some properties cannot be associated with a branch office through a member of staff.

Cont...

- We need to add the missing relationship which is called 'has' between the Branch and the Property entities.
- You need to therefore be careful when you remove relationships which you consider to be redundant.



Difference

Fan Trap:

A Fan Trap occurs when a single entity is in a one-to-many relationship with two or more other entities, **leading to ambiguity** in the pathway between certain entity occurrences.

Chasm Trap:

A Chasm Trap occurs when a model suggests the existence of a relationship between entity types, but the pathway between certain entity **occurrences is missing**, often due to optional relationships.

E(Enhanced)ER Modelling

- Complexity of the data make it difficult to use the traditional ER model.
- EER model provides more meaning than can be incorporated in the entity relationship model.
- To reduce this complexity of modelling:
 - improvements or enhancements to the existing ER model to make it able to handle the complex application in a better way.
- A diagrammatic technique for displaying the Sub Class and Super Class; Specialization and Generalization; Aggregation etc.
- It is based on developing the features for representing supertype/subtype relationships.

Generalization/Specialization

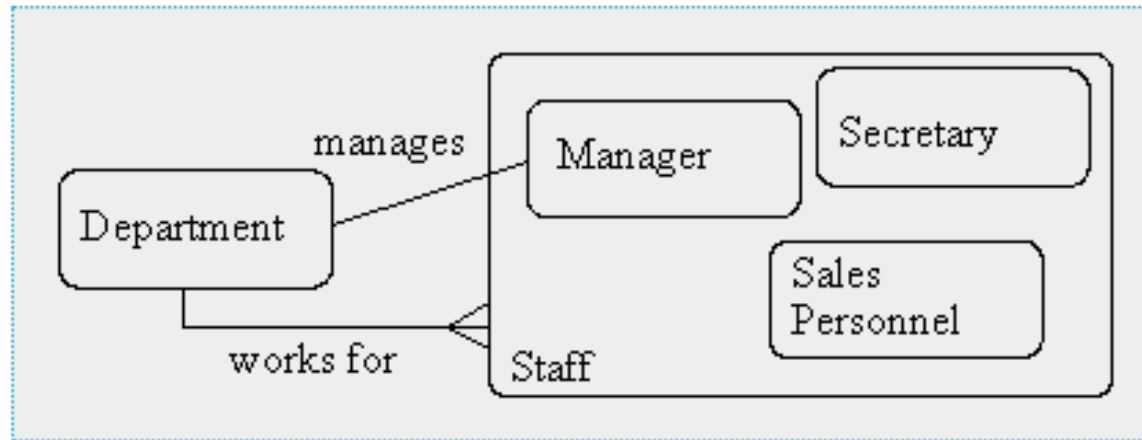
- Associated with special types of entities known as superclasses and subclasses.
- **Specialisation** is the process of defining a set of subclasses. There may be several specialisations of the same entity type
- **Specialized** classes are often called subclass while a **generalized** class is called a superclass.

Superclasses and Subclasses

- An entity type represents a set of entities of the same type such as *Staff*, *Branch*, and *PropertyForRent*. We can also form entity types into a hierarchy containing superclasses and subclasses.
- **Superclass:** An entity type that includes one or more distinct subgroupings of its occurrences, which require to be represented in a data model:
- **Subclass:** A distinct subgrouping of occurrences of an entity type, which require to be represented in a data model.
- For example, the entities that are members of the *Staff* entity type may be classified as *Manager*, *SalesPersonnel*, and *Secretary*. In other words, the *Staff* entity is referred to as the **superclass** of the *Manager*, *SalesPersonnel*, and *Secretary* **subclasses**.

Specialization

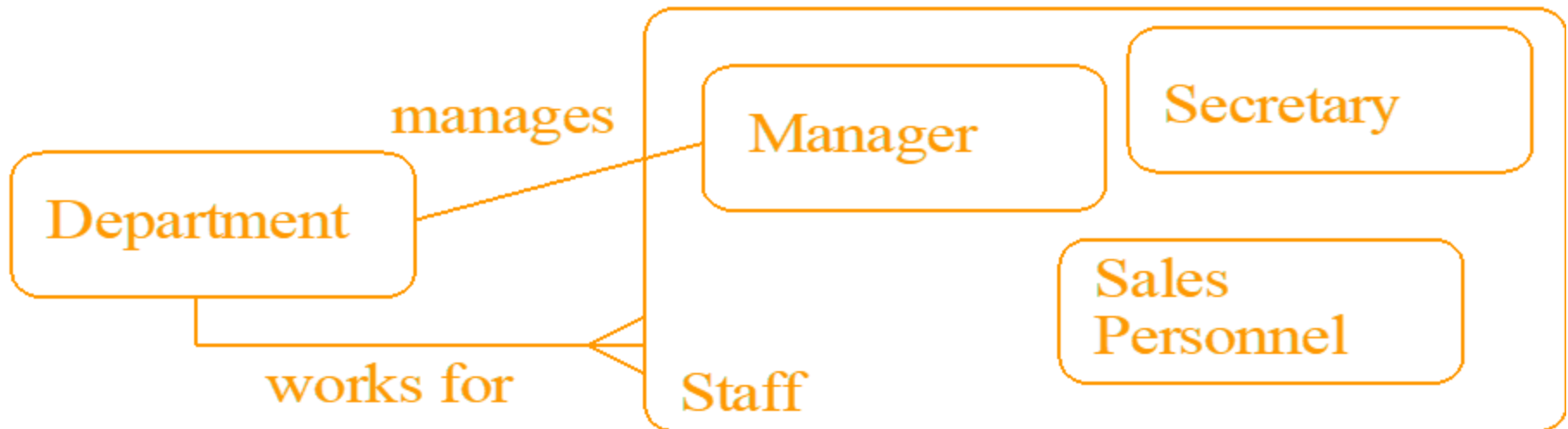
This is the process of maximizing the differences between members of an entity by identifying their distinguishing characteristics.



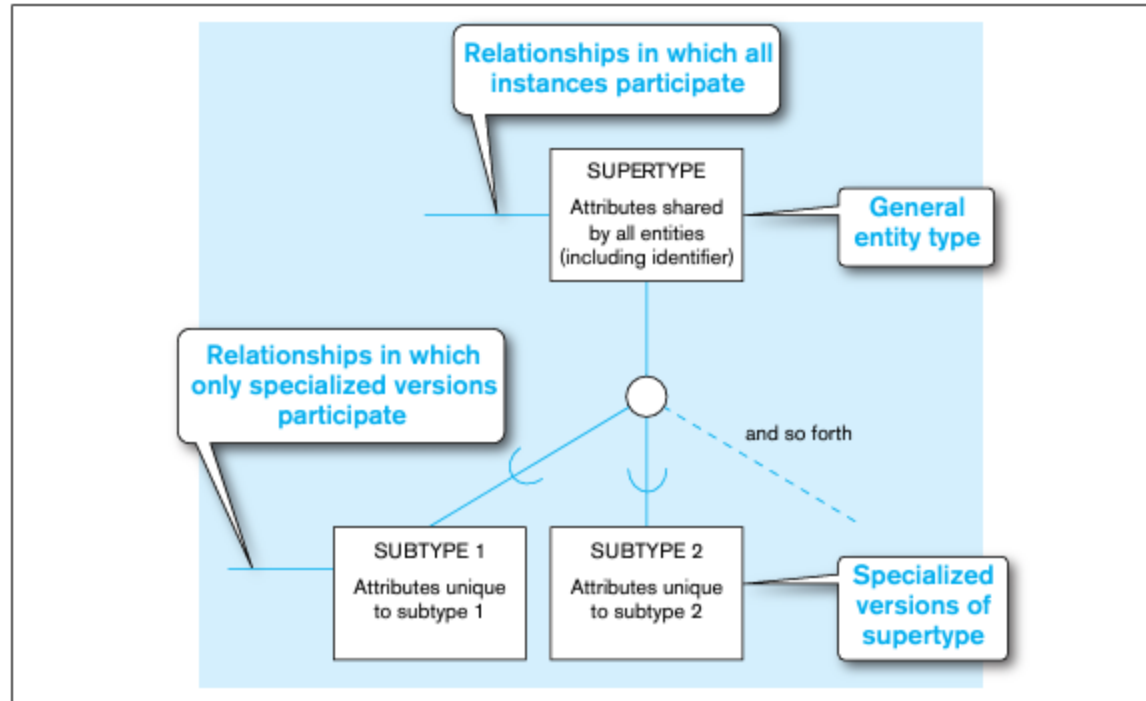
- Staff(staff_no, name, address, dob)
- Manager(bonus)
- Secretary(wp_skills)
- Sales_personnel(sales_area, car_allowance)

Cont..

- Here we have shown that the **manages** relationship is only applicable to the **Manager** subclass, whereas the **works_for** relationship is applicable to all staff.
- It is possible to have subclasses of subclasses.



Basic notations



Generalisation

- Generalisation is the process of minimising the differences between entities by identifying common features.
- This is the identification of a generalised superclass from the original subclasses. This is the process of identifying the common attributes and relationships.

Cont..

For instance, taking:

car(regno, colour, make, model, numSeats)

motorbike(regno, colour, make, model, hasWindshield)

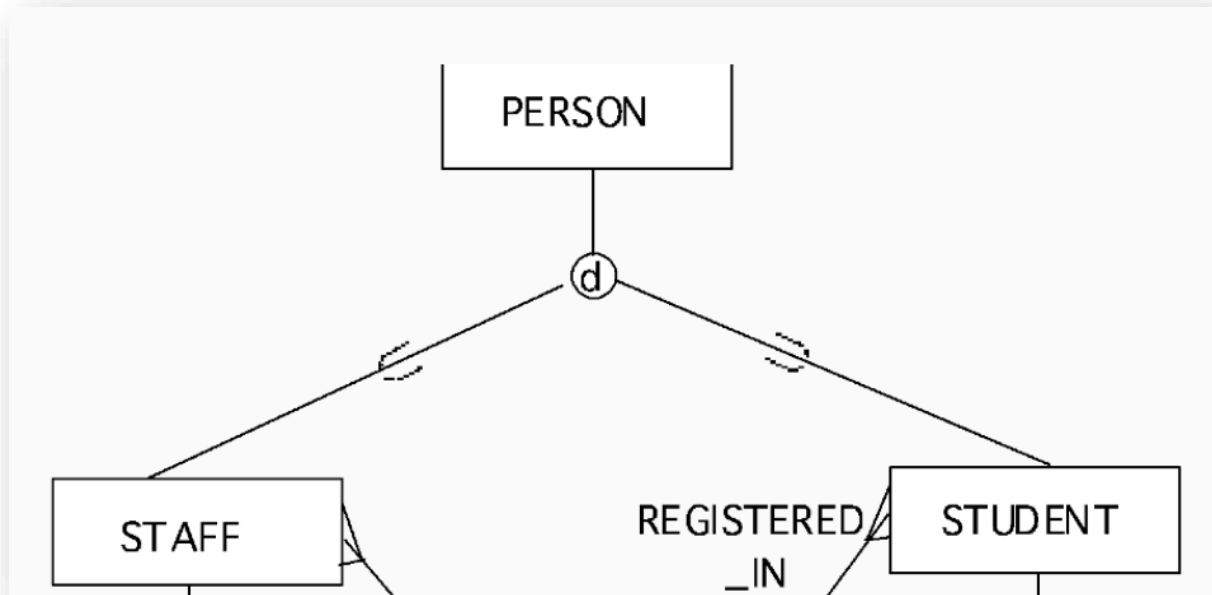
And forming:

vehicle(regno, colour, make, model, numSeats, hasWindshield)

In this case vehicle has numSeats which would be NULL if the vehicle was a motorbike, and has hasWindshield which would be NULL if it was a car.

Converting EER diagrams

- There are three possibilities for converting super-type/sub-type entities to a relational schema (logical model).



Option 1

- Create separate tables for the super-type and sub-type entities. Post the identifiers from the super-type to the sub-type.
- Advantage: no null values
- Disadvantage: many tables

Person

<u>ni_number</u>	fname	lname
YL-91-23-89-E	Scott	Free
JL-81-73-89-F	Grant	McPhail
JK-89-75-99-G	John	Lee

Staff

<u>staff_number</u>	ni_number
529015	JK-89-75-99-G
529028	JL-81-73-89-F

Student

<u>student_number</u>	ni_number
8929254	YL-91-23-89-E

Research Assistant

<u>staff_number</u>	grant_number
529028	GRF35869

Option 2

- Create a single relation for each subtype. Post the attributes from the super-type to each of the sub-types concerned.
- Advantage: no null values
- Disadvantage: many tables

Student

<u>student_number</u>	ni_number	fname	lname
8929254	YL-91-23-89-E	Scott	Free

Staff

<u>staff_number</u>	ni_number	fname	lname
529015	JK-89-75-99-G	John	Lee

Research Assistant

<u>staff_number</u>	grant_number	ni_number	fname	lname
529028	GRF35869	JL-81-73-89-F	Grant	McPhail

Option 3

- Create a single relation that has the union of all the attributes from the super-type and each of the sub-types. Add in a separate attribute to denote the type of each entity instance.
- Advantage: single table
- Disadvantage: many nulls

Person

<u>ni_number</u>	fname	lname	student_number	staff_number	grant_number	type
YL-91-23-89-E	Scott	Free	8929254	~	~	1
JL-81-73-89-F	Grant	McPhail	~	529028	GRF35869	2
JK-89-75-99-G	John	Lee	~	529015	~	3

Staff view

ni_number	fname	lname	staff_number
-----------	-------	-------	--------------

Research Assistant view

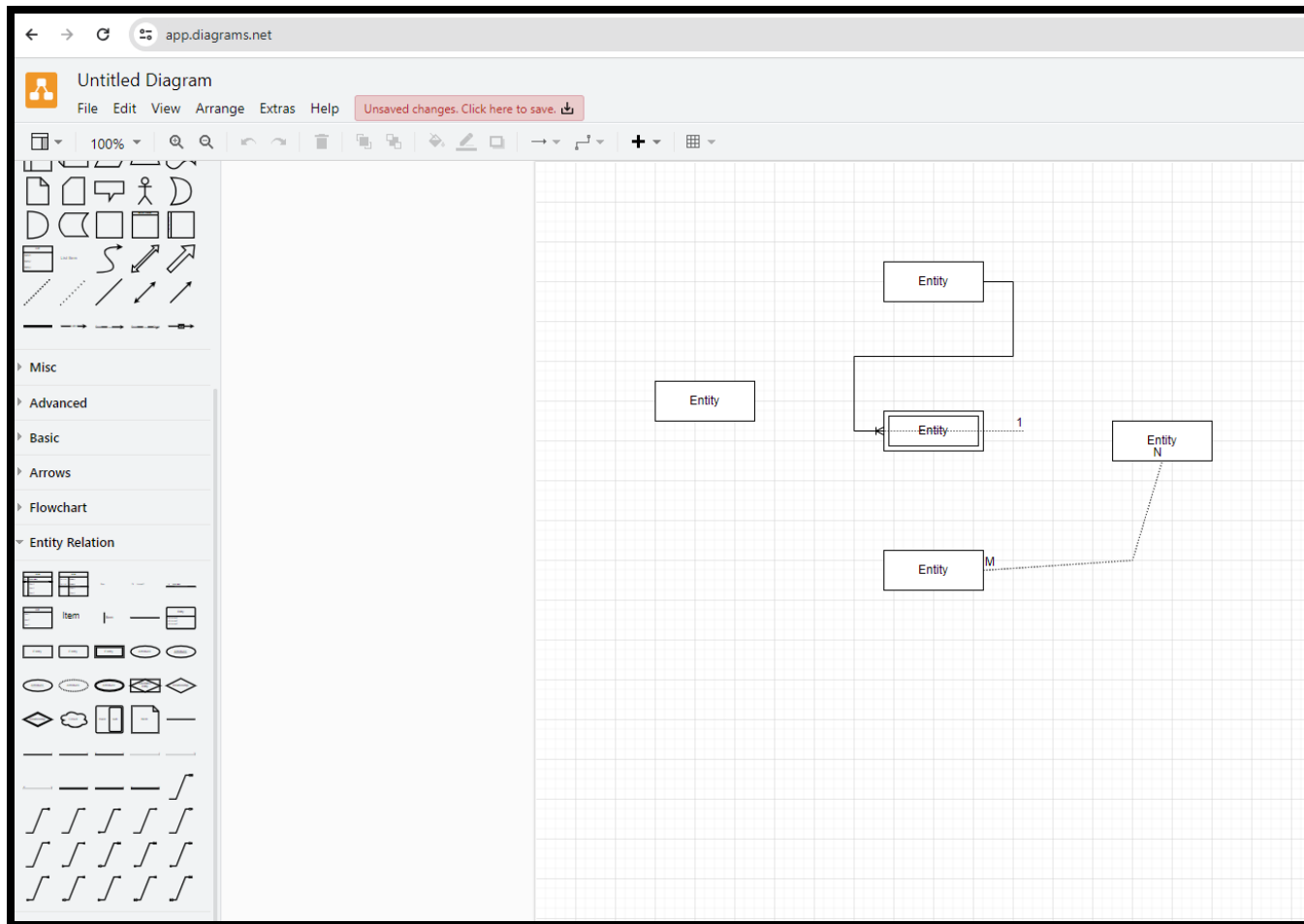
ni_number	fname	lname	staff_number	grant_number
-----------	-------	-------	--------------	--------------

Student view

ni_number	fname	lname	student_number
-----------	-------	-------	----------------

Drawing ERD

You may use pen and pencils or **draw.io** website



Home work: Activity

- Inside a Google data center <https://www.youtube.com/watch?v=XZmGGAbHqa0>
- Microsoft data center under water <https://www.youtube.com/watch?v=Ag2L5RY9CHI>

Questions

Course Content

1. Introduction to Relational Databases (*Introduction + Relational Model*)
2. Data Modelling - (*Entity Relationship Modelling + The Enhanced Entity Relationship Model*)
3. Database Design and SQL - (*Logical modelling + Introduction to SQL*)
4. Further SQL - (*Advanced SQL queries + Creating tables with SQL*)
5. Normalisation - (*Normalisation to second normal form + Third normal form*)

Database Fundamentals – CS990

Database and Web Systems Development - CS952