COMM1822

Term 2 2022

Introduction to Databases for Business Analytics

Week 2 Entity Relatio (ER) Modelling Part 2

Lecturer-in-Charge: Kam-Fung (Henry) Cheung
Email: kf.cheung@unsw.edu.au
Tutors: Theresa Tran (Tutor-in-Charge)
Liam Li Chen
Kathy Xu
PASS Leader: Srilekha Chandrashekara Kolaki

UNSW SYDNEY

Copyright

- There are some file-sharing websites that specialise in buying and selling academic work to and from university students.

- If you upload your original wor other student downloads and presents it as their own eit **ight be found guilty of collusion — even years after graduatio**

- These file-sharing websites may also accept purchase of course materials, **such as copies of lecture slides and tutorial handouts. By law, the copyright on course materials, developed by UNSW staff in the course of their employment, belongs to UNSW. It constitutes copyright infringement, if not academic misconduct, to trade these materials.**

Acknowledgement of Country

UNSW Business School acknowledges the Bidjigal (Kensington campus) and Gadigal (City campus) the traditional custodians of the lands where each campus is located.

We acknowledge all Aboriginal and Torres St Islander Elders, past and present and thei communities who have shared and practiced their teachings over thousands of years including business practices.

We recognise Aboriginal and Torres Strait Islander people's ongoing leadership and contributions, including to business, education and industry.

UNSW Business School. (2022, May 7). *Acknowledgement of Country* [online video]. Retrieved from https://vimeo.com/369229957/d995d8087f

Agenda

Entity Relationship Modelling (from week 1)

- Recap
- Weak Entity

Advanced Entity Relatio

- Supertype
- Subtype

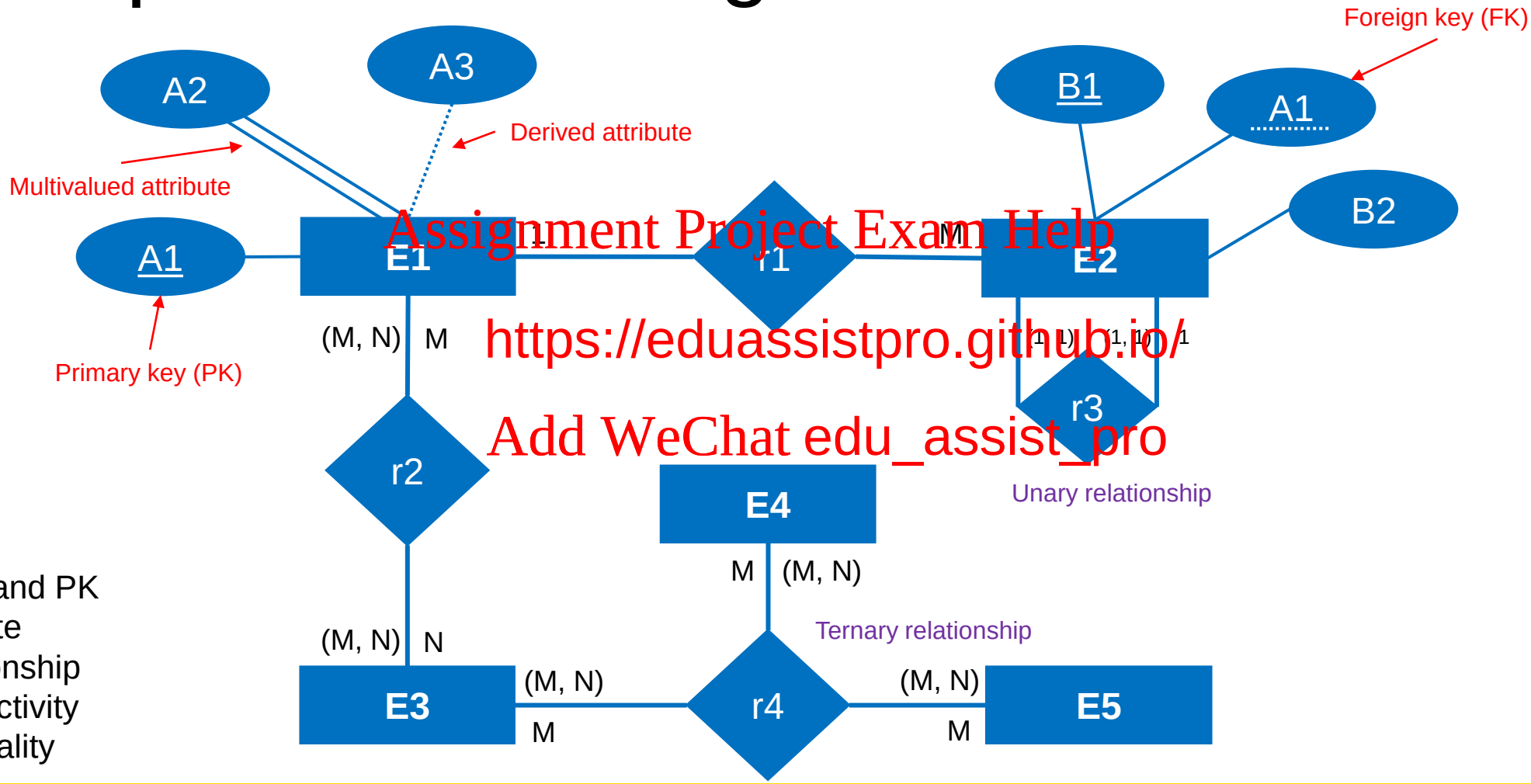
Convert from an Entity Relationship Model to a Relational Model

Recap: ER Modelling 1

- **Data Modelling:**
  - Data model as a (relatively) simple abstraction of the complex real-world.
  - One modelling technique to design a database: Entity Relationship Modelling.

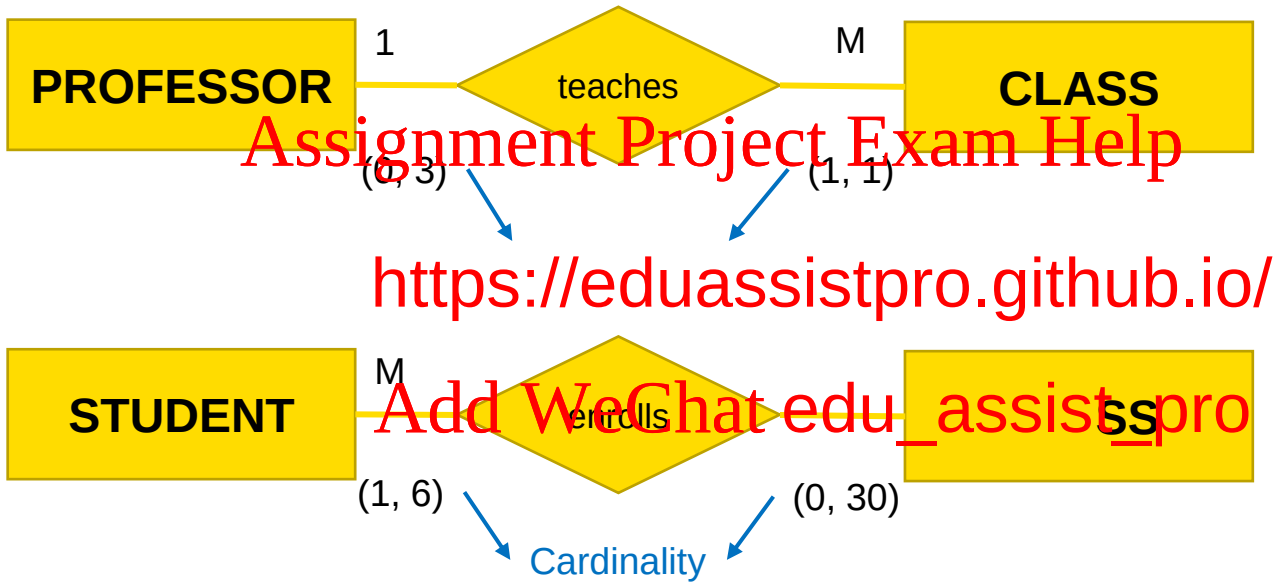
- **Entity Relationship Mod**
  - Entity Types and Entity Instances
  - Attributes and Values
  - Keys
  - Relationships
  - Connectivity and Cardinality

Recap: ER Modelling 1



- Entity and PK
- Attribute
- Relationship
- Connectivity
- Cardinality

Connectivity and Cardinality



One-to-Many Relationship

Many-to-Many Relationship

How to read this?

- A professor teaches (0, 3) classes. A class is taught by (1, 1) professors.
- A student enrolls in (1, 6) classes. A class has enrolled in it (0, 30) students.

Existence Dependence and Independence

Existence Dependence and Independence

Existence dependence: Entity exists in the database only when it is associated with another related entity occurrence

e.g., parents

Existence independence: Entity exi om all of its related entities, and referred to as a **strong entity** or **regular entity**

e.g., customer & product in a supermarket

Weak (Non-identifying) Relationship

Primary key of the related entity does not contain a primary key component of the parent entity.

CRS_CODE is a primary key of C
CRS_CODE is a foreign key of Class table.

CRS_CODE is a foreign key, but it is not part of the primary key of Class table.

Strong (Identifying) Relationship

Primary key of the related entity contains a primary key component of the parent entity.

CRS_CODE is a primary key of Course table, and CRS_CODE is a foreign key of Class table.

CRS_CODE is also part of the primary key of Class table. CRS_CODE is part of the composite primary key for Class table.

Weak Entity

Weak Entity

- **Weak entity** is an entity that **relies on the existence of another (strong or independent) entity**. It has a primary key (PK) that is partially or totally derived from the parent entity in the relationship.

rents & children

- Weak entity meets two con

  - **Existence-dependent:** Cannot exist without en it has a relationship.
  - Has primary key that is **partially or totally dari parent entity** in the relationship.

- Database designer usually determines whether an entity can be described as weak based on the business rules.

Example of a Weak Entity in an ERD

- A child must have one parent working in the company.
- If both parents work in the company, you only have to connect to one. E.g., UNSW childcare: it is connected to the parent who will pay childcare fees ☺

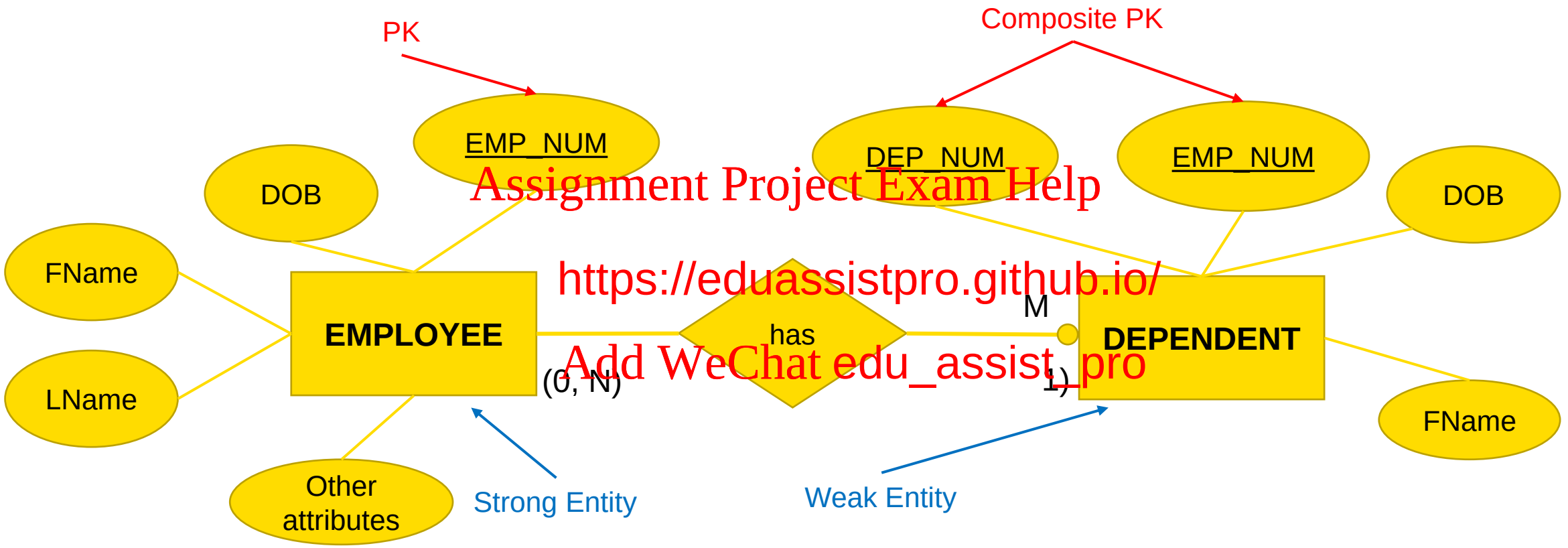
Weak Entity: DEPENDENT

As for fulling the conditions:

a. **Existence-dependent:** Cannot exist without entity with which it has a relationship – in this case it is the DEPENDENT entity. A child must exist with one of his/her parents.

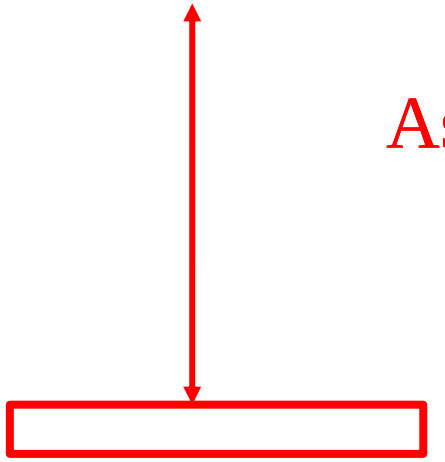
b. Has primary key that is **partially or totally derived from the parent entity** in the relationship – in this case, EMP_NUM in DEPENDENT entity is associated with EMP_NUM of the EMPLOYEE table.

Example of a Weak Entity (with Attributes)



Existent-Dependent Relationship between EMPLOYEE & DEPENDENT

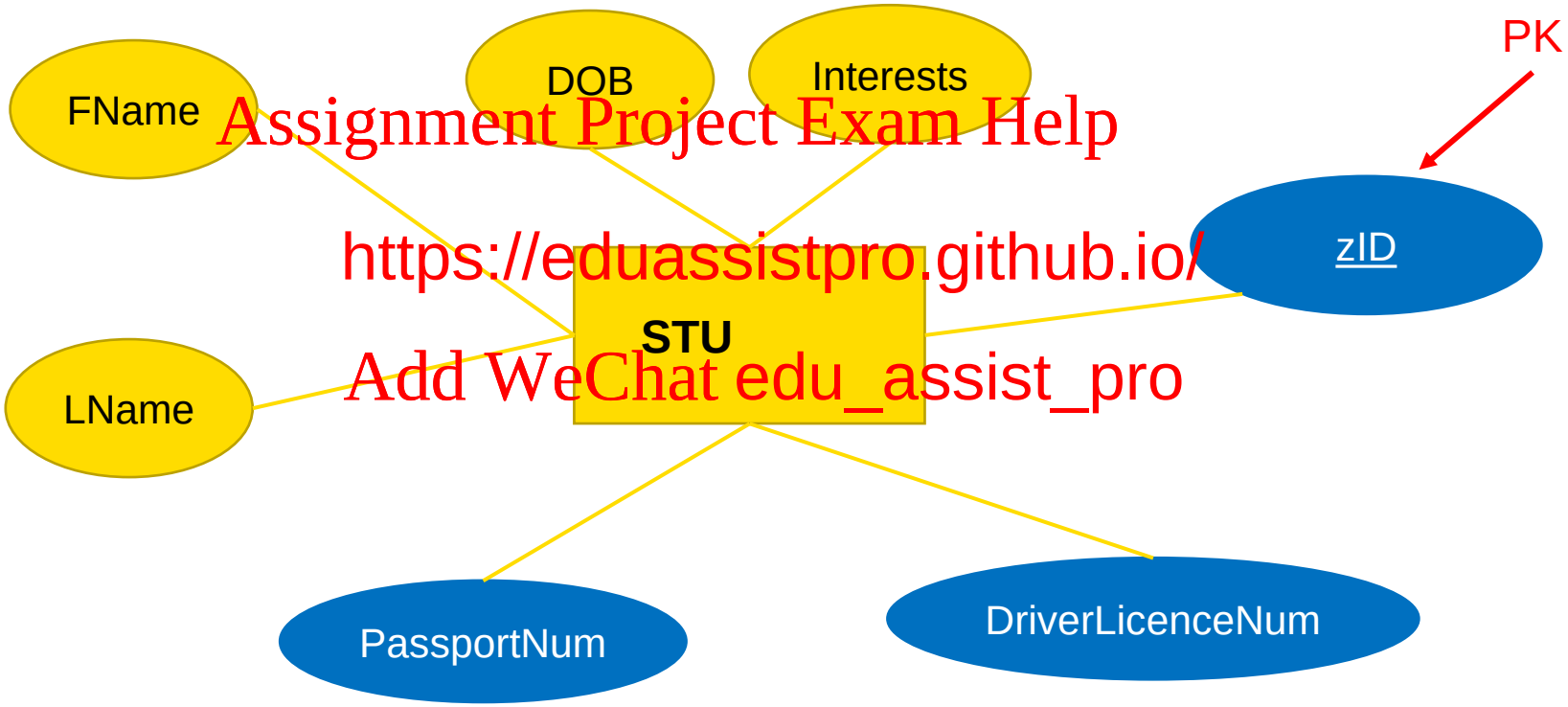
Example of a Weak Entity



Desirable Primary Key Characteristics

Unique value	**Cannot be null**
Non intelligent	**Should not have embedded semantic meaning, e.g., use zID as PK rather than name**
No change over ti	**nge**
Preferably single-attribute	**Remember attributes m / mecomes foreign key, multiple nk tables**
Preferably numeric	**To avoid typing errors; Can use auto-increment, e.g., zID**
Security-compliant	**Using Social Security Number (SSN) as a SID is a bad idea.**

Candidate Key and Primary Key



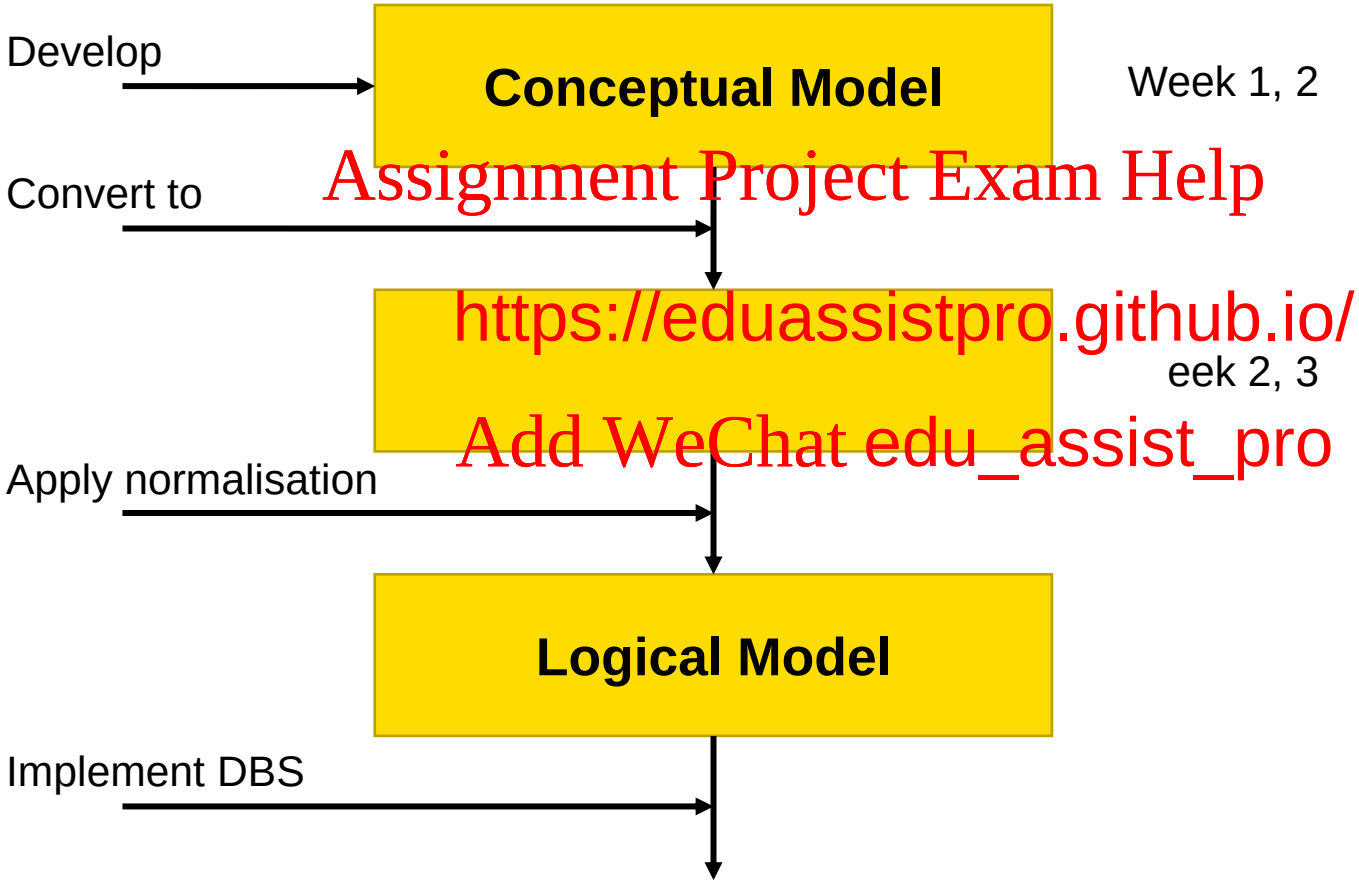
Candidate Keys:

- zID
- PassportNum
- DriverLicenceNum

Plan: ER Modelling 2

- **Enhanced Entity Relationship Modelling**
  - **Composite entity** (bridge entity)
  - **Supertype** and **su**
  - **Generalisation** an
  - **Constraints** (completeness, disj
- **Exercise**

Database Design Process Modelling



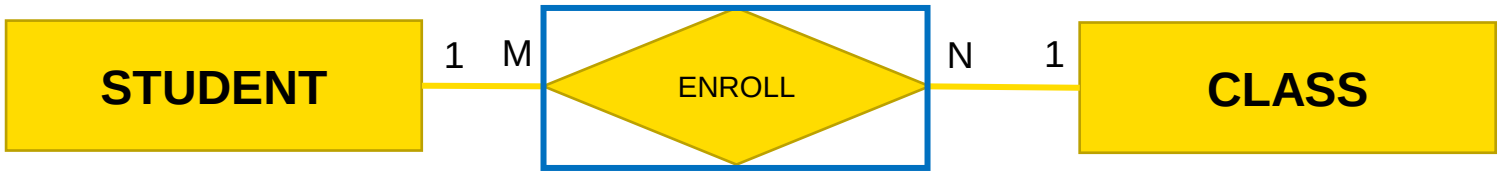
Composite Entity

Composite Entity

A **composite entity** (bridge entity, associative entity) is an entity type that associates the instances of one or more entity types. It contains attributes that are peculiar (singular) to the relationship between those entity instances.

- The composite entity builds a **bridge** between the original entities.
- The composite entity is composed of the **PKs of the original entities**.
- The composite entity may cont

A regular M:N Relationship



A composite entity dealing with M:N Relationship

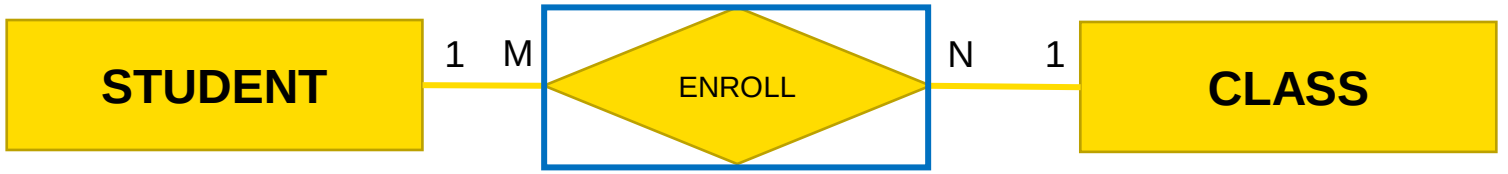
Composite Entity

- **M:N relationships** (many-to-many relationships) should be **avoided**.
- **Relational databases** can only handle **1:M relationships** (one-to-many relationships).
- **M:N relationships** should be **decomposed** (broken down) to 1:M relationships by creating a **composite entit**
- The composite entity builds al entities.
- The composite entity is composed of the **PKs** **al entities**.
- The composite entity is **existence-dependent** on the original entities.
- The composite entity may contain **additional attributes.**

Composite Entity

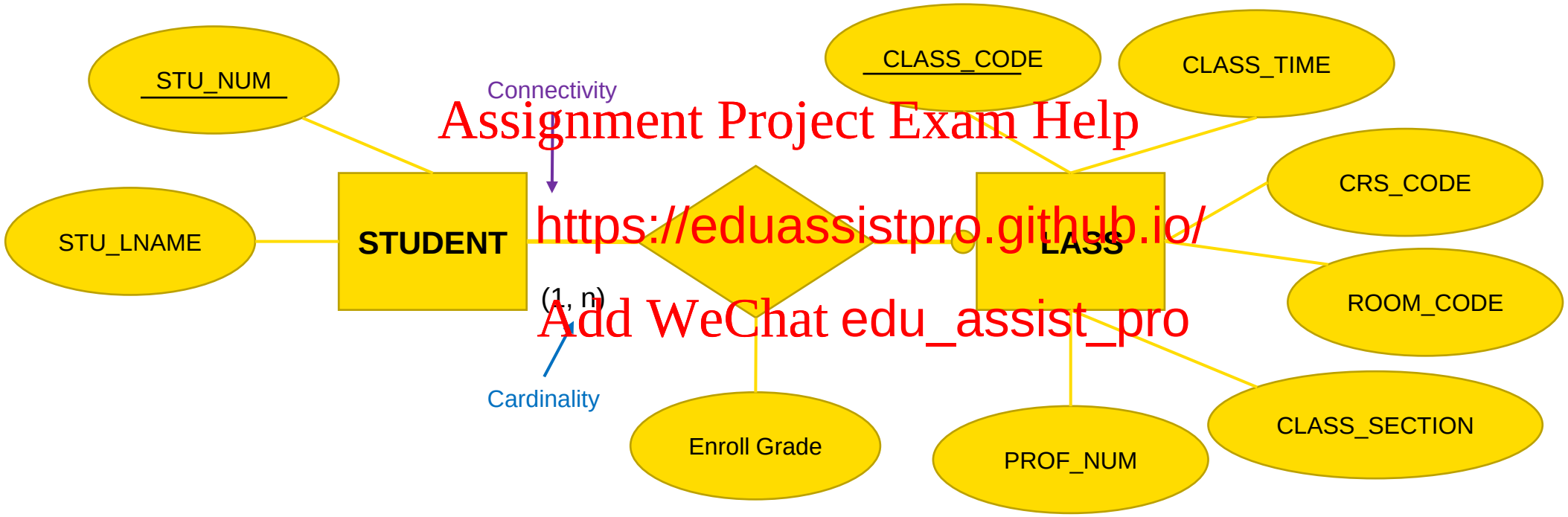
- **Relational databases** can only handle **1:N relationships** (one-to-many relationships) or **1:1 relationships.**
- **M:N relationships** (many-to-many relationships) should be **avoided (via building composite entity)**.
- **A M:N relationship** shoul en down) to **two** 1:M relationships by creating

A regular M:N Relationship

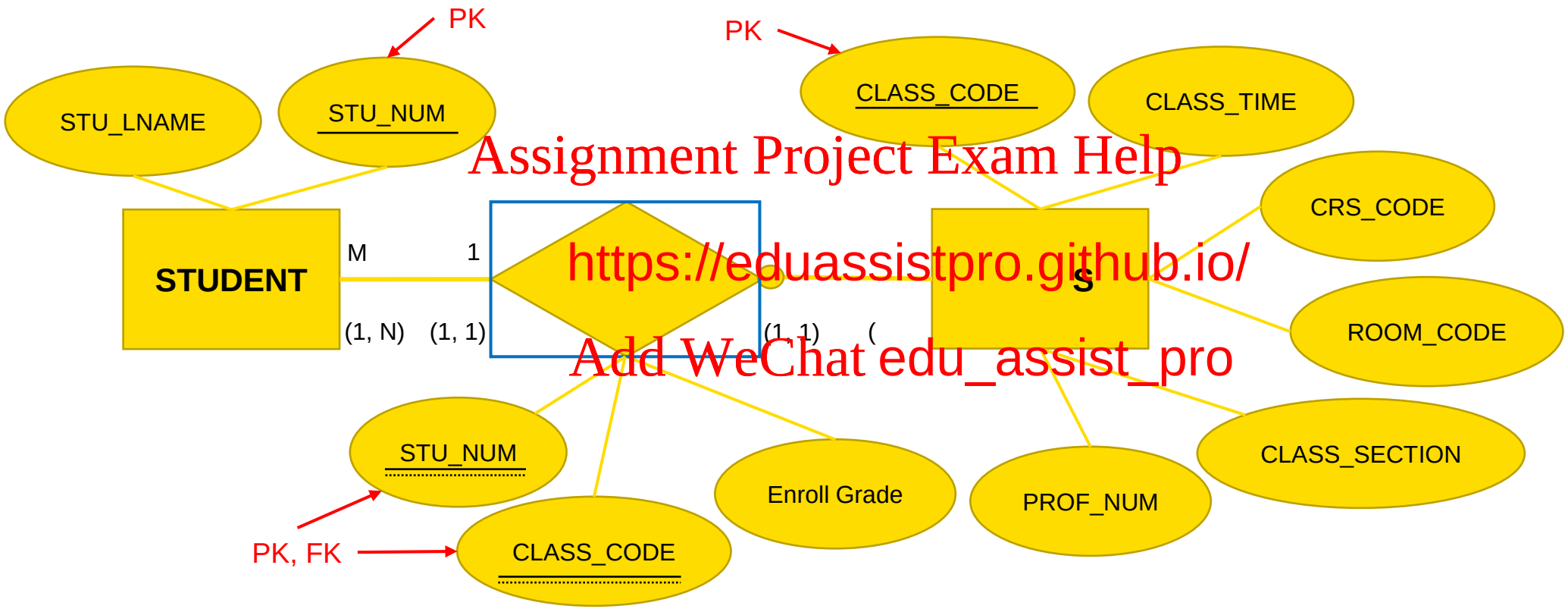


A composite entity dealing with M:N Relationship

Without Composite Entity (Original)



With Composite Entity



Inappropriate Approach

STUDENT

STU_NUM	STU_LNAME	CLASS_CODE
001	Brown	1602, 1603
002	Pink	1602, 2603
003	Green	5992
004		3

CLASS

CLASS_CODE	CRS_CODE	
1602	CIS-200	
1603	CIS-300	510
2603	SAD-100	240
5992	GM-200	350
5993	ADB-300	120

STUDENT

STU_NUM	STU_LNAME
001	Brown
002	Pink
003	Green
004	White

Correct Approach

ENROLS

STU_NUM	CLASS_CODE
001	1602
001	1603
002	1602
002	2603
03	5992
04	5992
	5993

CLASS

CLASS_CODE	ClassName	PROF_N
1602	CIS-200	700
1603	CIS-300	510
2603	SAD-100	240
5992	GM-200	350
5993	ADB-300	100

Ternary Relationship Without Composite Entity

Ternary Relationship With Composite Entity

- ✓ A guest can be included in many schedule records;
- ✓ A schedule record includes only one guest;
- ✓ A room can be included in many schedule records;
- ✓ A schedule record includes only one room;
- ✓ An event can have many schedule records;
- ✓ A schedule record is only for one event.

Chapter 5

d Data Modeling

Supertype and Subtype

UNSW SYDNEY

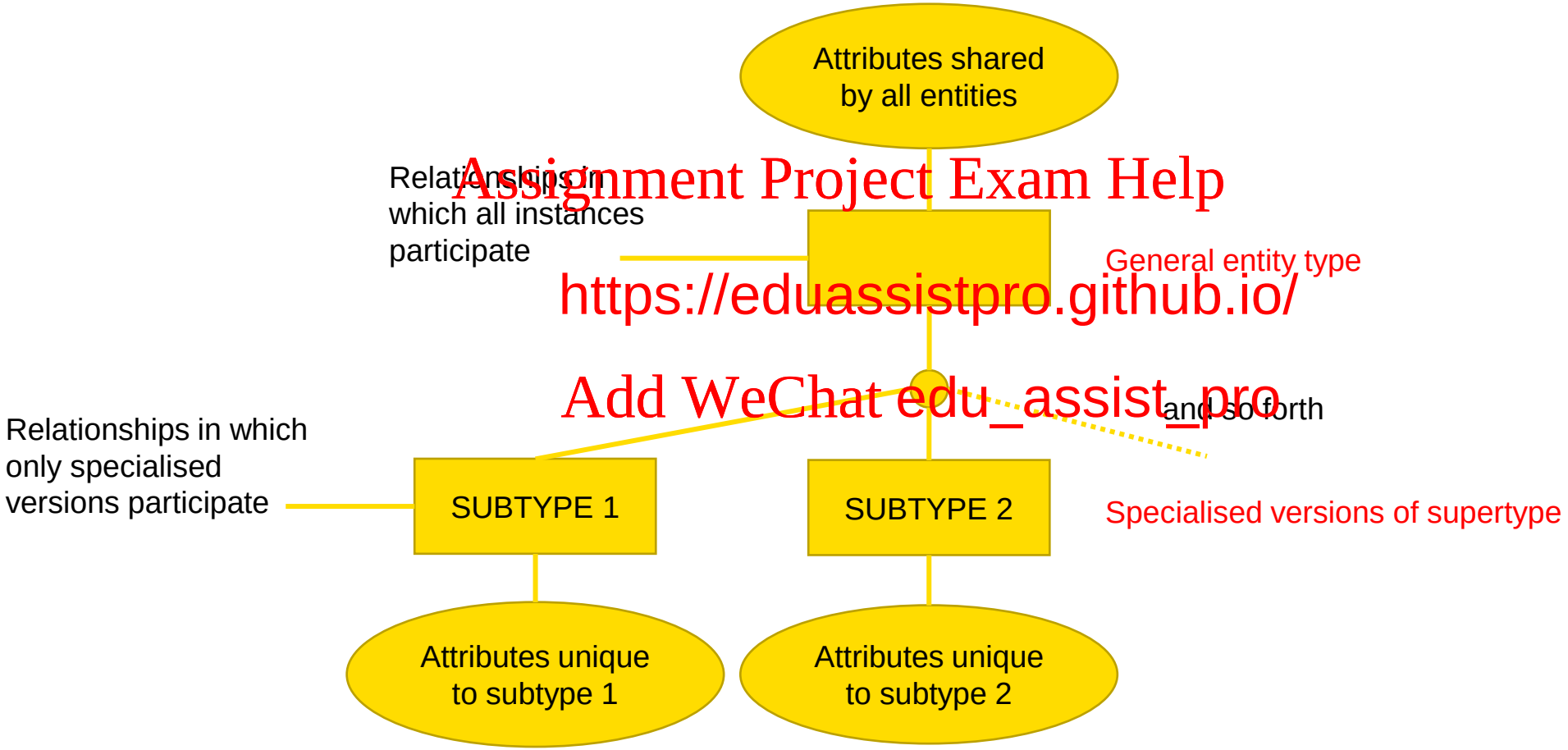
Supertype and Subtype

"A **supertype** is a more **generic** entity type compared to its subtypes."

"A **subtype** is a more **specific** entity type compared to its supertype."

- A subtype entity inherit **type.**
- A subtype has additional, **specific att**
- **An instance (occurrence) of a subtype** is also **an instance (occurrence) of the supertype.**
  (The other way around, an instance of the supertype may or may not be an instance of one or more subtypes.)

Supertype and Subtype



Example of Supertype and Subtype

Explain later

Generalisation and Specialisation

UNSW SYDNEY

Generalisation and Specialisation

- **Generalisation**: The process of defining a general entity type from a set of specialised entity types. It is a **bottom-up** process from subtypes to supertypes.

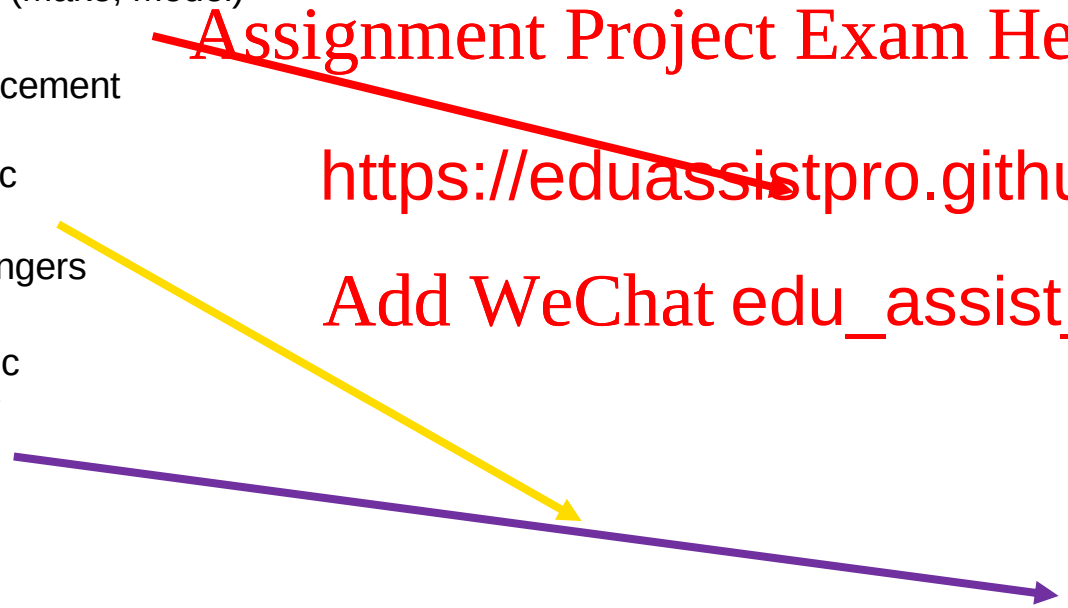
- **Specialisation**: The process of defining one or more subtypes of the supertype. It is a **top-down** process from supertypes to subtypes.

Generalisation

Question: How to generalise the listed two entity types: CAR and TRUCK?

Generalisation

- What are the common attributes?
  - Vehicle_ID
  - Vehicle_Name (Make, Model)
  - Price
  - Engine Displacement
- What are the specific attributes for Car?
  - No_of_Passengers
- What are the specific attributes for Truck?
  - Capacity
  - Cab_Type



Specialisation

How to specialise the entity type PART??

Specialisation

Specialisation

How to specialise the entity type Staff?

Specialisation

Constraints

Constraints

- The **Completeness Constraint** describes whether an instance of a supertype must also be an instance of at least one subtype.

- The **Disjointness Con**

er an instance of a supertype may simultaneously be a two (or more) subtypes.

Constraints

- A staff *must* be either an academic staff **or** admin staff (total
- An staff ***may* also** be an ad

Constraints

Common sense tells us:

A vehicle *may* be a car or truck or something else (partial specialisation): single line.

A car *must not (CANNOT)* be a truck (disjoint rule): d.

Constraints

- A part *must* be purchased or manufactured (total specialisation rule): double line.
- A manufactured part *may* also be a purchased part (overlap rule): o.

Completeness Constraint

- Specifies whether each supertype occurrence must also be a member of at least one subtype

- Types
  - **Partial completeness**: Not every sup rence is a member of a subtype
  - **Total completeness**: Every supertype occurrence must be a member of any subtype

VEHICLE such as Bus (which is not listed out)

STAFF – Academic Staff & Admin Staff

Disjoint and Overlapping Constraints

- **Disjoint subtypes**: Contain a unique subset of the supertype entity set VEHICLE – Car vs Truck
  - Known as **nonoverlapping subtypes**
  - Implementation is bas _____ type discriminator attribute in the supertype
- **Overlapping subtypes**: Contain _____ subsets of the supertype entity set STAFF – Academic Staff & Admin Staff
  - Implementation requires the use of one discriminator attribute for each subtype

Subtype Discriminator(s)

- **Subtype discriminator(s)** are the attribute(s) of the supertype that determine (code, note, identify) the target subtype.
  - Disjoint Constraint Rule: One attribute.
  - Overlapping Constraint veral attributes.

e attribute (Vehicle_Type) with
ossible values indicates the
disjoint rule).

“C” “T”

Vehicle_Type	Comment
C	The VEHICLE is a CAR.
T	The VEHICLE is a TRUCK.
Null	The VEHICLE is neither a CAR nor a TRUCK.

A composite attribute (Staff_Type) with sub-attributes (with “Yes”/“No” values) indicates the subtype (overlap rule).

Staff_T		Inment
Academic Staff	Admin Staff	
Yes	No	The Staff is a member of the Academic Staff subtype.
No	Yes	The Staff is a member of the Admin Staff subtype.
Yes	Yes	The Staff is both an Academic Staff and an Admin Staff.

How about No / No?

Specialisation Hierarchy Constraint Scenarios

A simple attribute (Employee_Type) with different possible values indicates the subtype (disjoint rule).

Can the Empolyee_Type be null?

ER Modelling Guideline

- The data items should be put into **an entity (logical group)**.
- For each entity, there should be a Primary **key** that uniquely identifies individual members of entity type.
- There should be **no re** del.
- Ask yourself the followi
  - What are the relevant entities here?
  - What are the relevant relationships h
  - Can I generalise some entities?
- Document your **assumptions** as you go.
- Leave connectivity and **cardinalities** until the end.

Questions

Source: canningtonvet.com.au

Public Holiday Arrangement in Week 3

- No lecture on Queen's Birthday 13 June (Monday)!
- A lecture recording of Week 3 will be uploaded on 14 June by 12 pm.

- Tutorials will go as usual!

Exercise

Draw an ER diagram for this situation (state any assumptions that you make). Based on the ER diagram, draw the relevant relational model.

- The firm has a number of sales offices in several states. Attributes of sales office include Office_number (identifier) and Location.
- Each sales office is assigned o s of employee include Employee_ID (identifier) and Employee_Nam ed one only one sales office.
- For each sales office, there is a o manage that office. An employee may manage only the sales office to which he or s
- The firm lists property for sale. Attributes of properti erty_ID (identifier) and Location. Components of Location include Address, City, State, and Zip_Code.
- Each unit of property must be listed with one (and only one) of the sales offices. A sales office may have any number of properties listed or may have no properties listed.
- Each unit property has one or more owners. Attributes of owners are Owner_ID (identifier) and Owner_Name. An owner may own one or more units of property. An attribute of the relationship between property and owner is Percent_Owned.