Zawansowane Modelowanie i Analiza Systemów Informatycznych (l-2)

Polsko-Japońska Wyższa Szkoła Technik Komputerowych
Katedra Systemów Informacyjnych
2013

Overwiev

- **Conceptual schema representation**
- **Some examples to illustrate the principles of ORM sentence construction.**

Introduction to Object Role Modeling Conceptual Schema Representation

- **Any conceptual modelling method should**
  - stimulate thinking about information
  - clearly represent the semantics of the information to be modelled
  - assist to construct semantically correct information models
  - provide a simple transformation to relational technology, or any other data model/implementation

- **The Object-Role Modelling methodology is well positioned to to support the designer in perofrming these tasks**

What is information?

- ***Data - items stored in a database***
- ***Semantics - the meaning of that data.***

Information = Data + Semantics

The Construction of Elementary Facts In English And Their Representation

EXAMPLES :

1. ***Mary Smith works in Sales.***
2. ***Employees are identified by an employee number.***
3. ***Beethoven wrote the Pastoral symphony.***
4. ***Student 990001 received grade 6 for Database Design.***
5. ***David Brown works in Software Maintenance and has id CS01.***
6. ***Employee 1432 is supervised by Employee 2341.***

1. Mary Smith works in Sales.

Assume the universe of discourse is concerned with employees in a company and the information required is the department in which each employee works.

Consider the following example :

Mary Smith	*Sales*
David Hicks	*Sales*
Sally Wong	*Account*

The first line of the output example perhaps expresses

Mary Smith works in Sales.

„works”   ??????

How are the entity types represented?

A Type is a set of all possible instances, so each entity is an instance of a particular type.

EMPLOYEE and DEPARTMENT are represented by
EMPLOYEE_NAME and DEPARTMENT_NAME

The sentence becomes:

The Employee with
EMPLOYEE_NAME Mary Smith
works in
the Department with
DEPARTMENT_NAME Sales

Terminology cont. example

The Employee with	*Entity Type*
EMPLOYEE_NAME	*Label Type*
Mary Smith	*Label Instance*
works in	*Role*
the Department with	*Entity Type*
DEPARTMENT_NAME	*Label Type*
Sales	*Label Instance*

Semantically Equivalent sentence

the Department with

DEPARTMENT_NAME

Sales

Employs

The Employee with

EMPLOYEE_NAME

Mary Smith

The existence of the role works in necessarily implies the existence of the role employs.

The two roles are said to belong to a FACT TYPE

We will call this fact type for instance; Allocation.

Type declarations:

Entity types

Label types

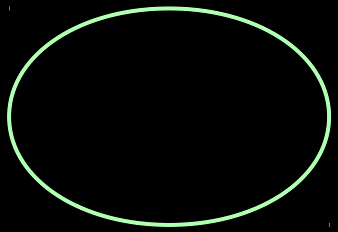
Fact types (and their roles)

are expressed at the conceptual schema.

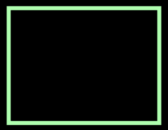
instances (at a point in time) are stored in the database.

- ***NOTE: do not confuse an instance with a type!***

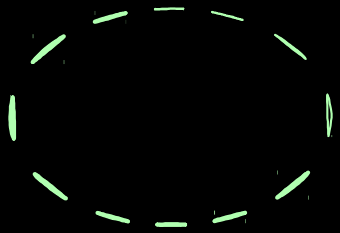
Basic ORM Notation



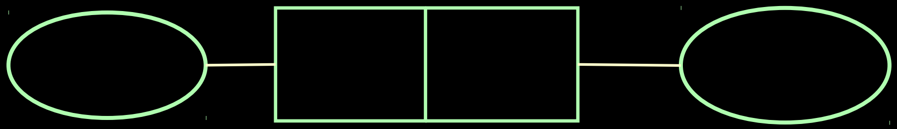
Entity type or Object type



Role

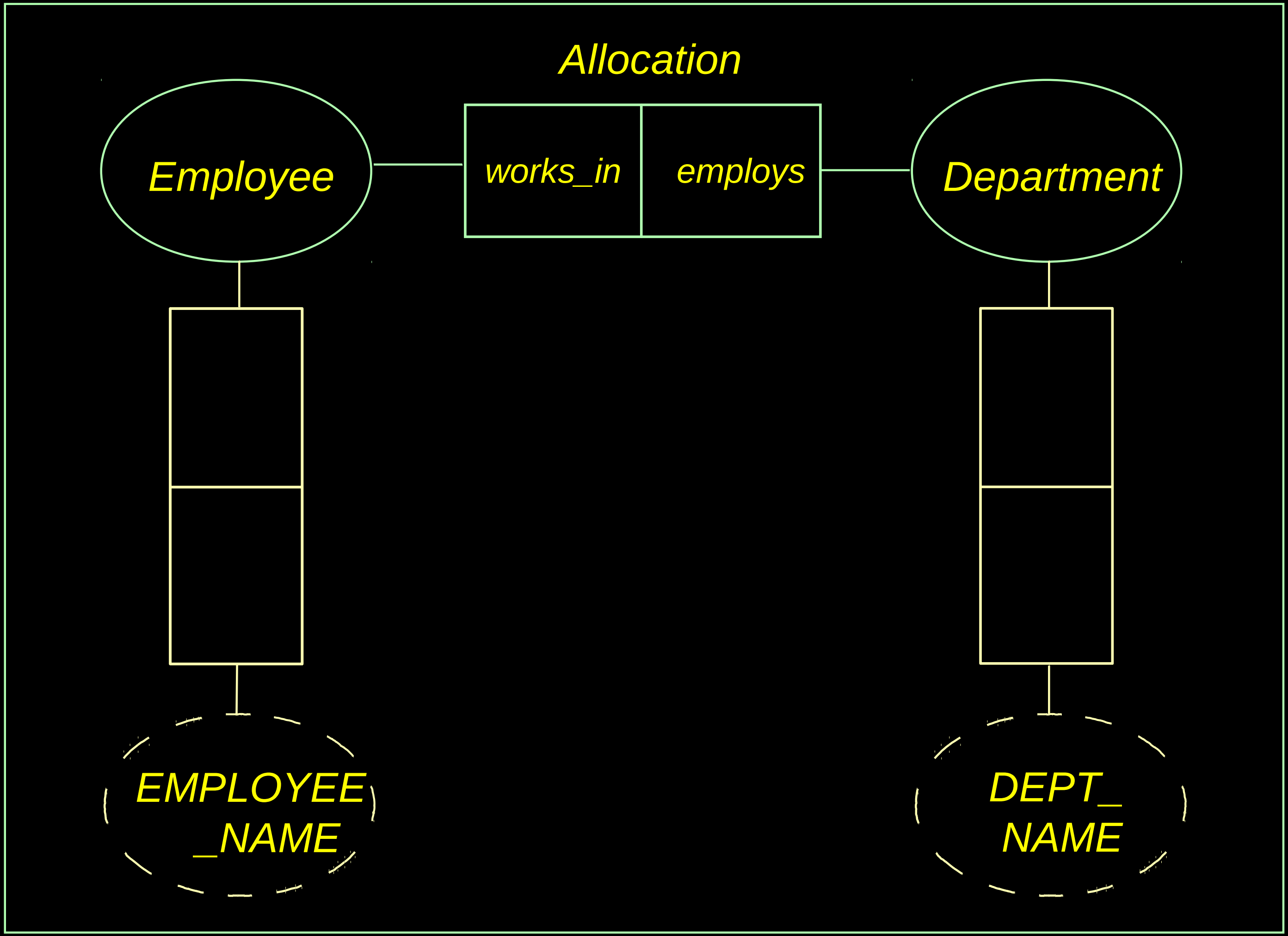


Label type

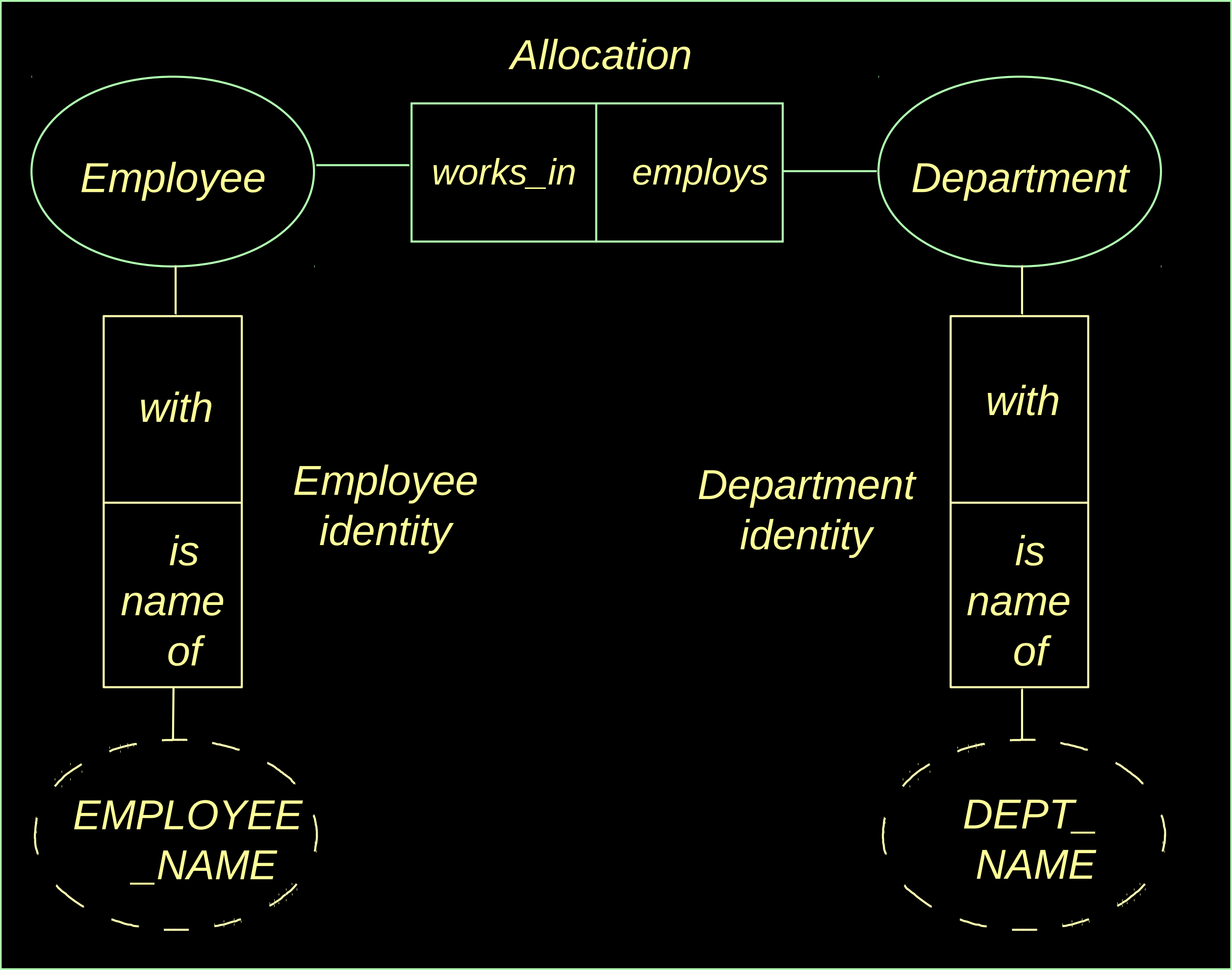


Fact type (binary)

Graphical representation of the sentence

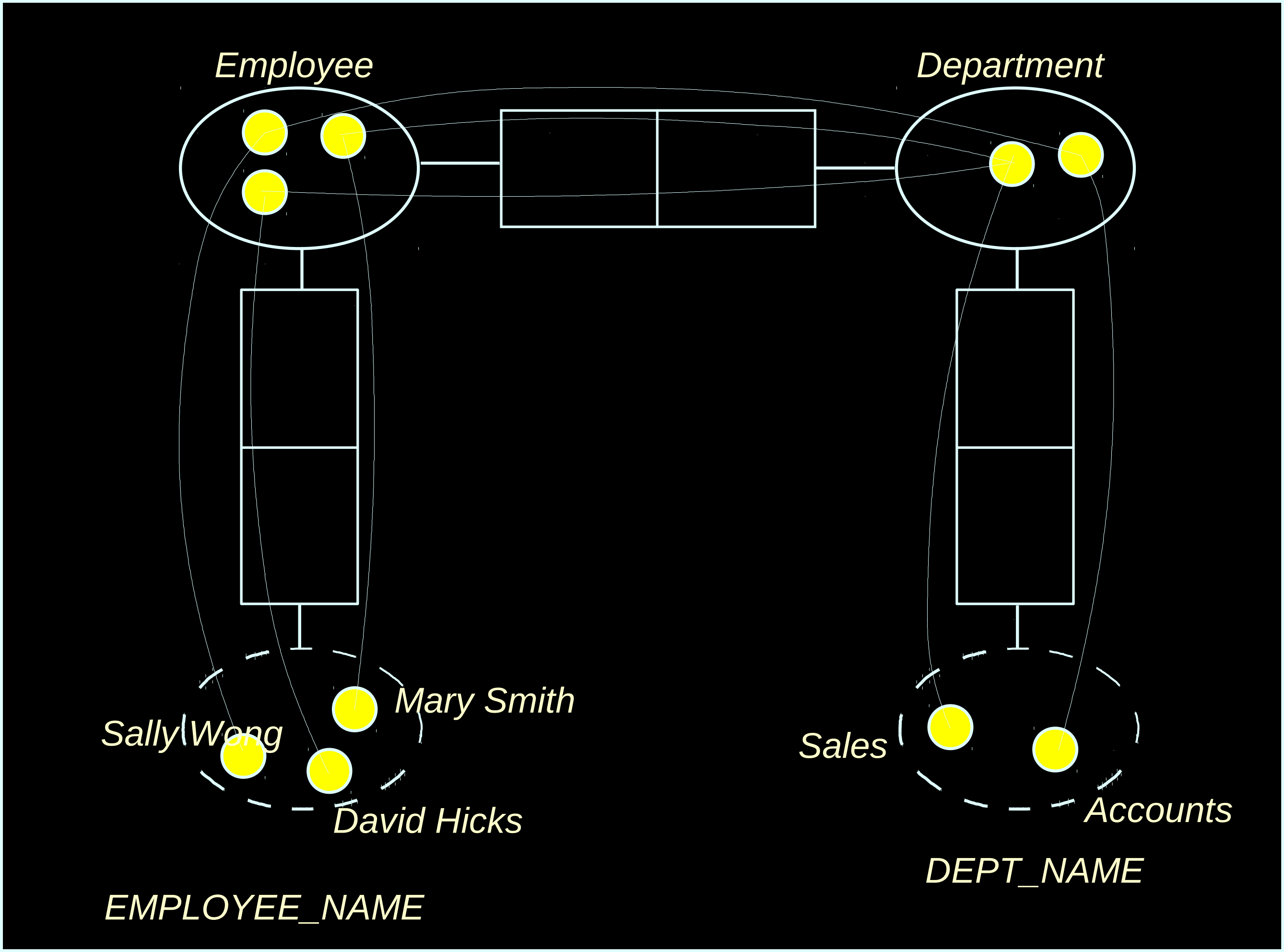


Additional fact types in the diagram



Employee identity and Department identity are also fact types.

Using a diagram to show the information base



A tabular representation

Employee	*Department*
EMPLOYEE _NAME	*DEPT_NAME*
works_in	*employs*
Mary Smith	*Sales*
Sally Wong	*Accounts*
David Hicks	*Sales*
etc	*etc*

Conceptual Schema

Database

Points for discussion

We want to make our databases as semantically meaningful

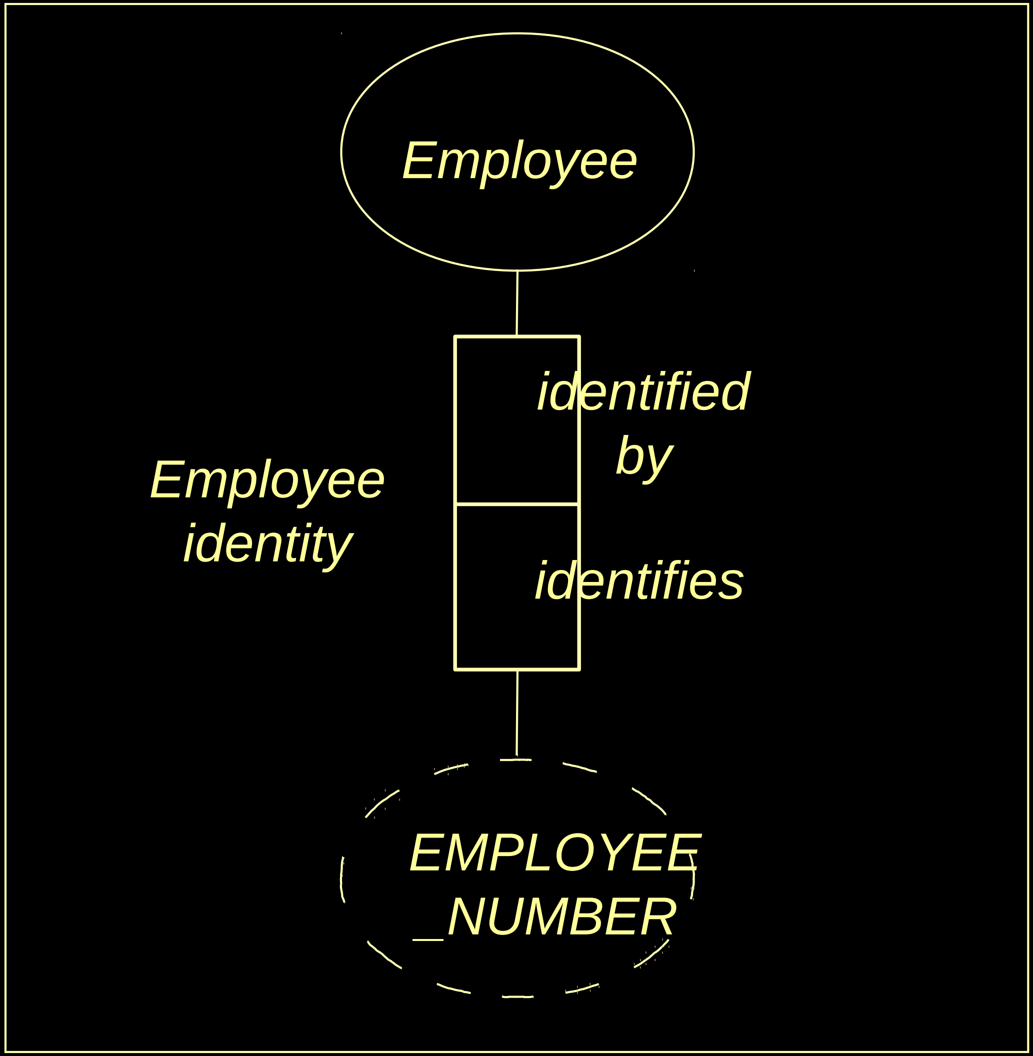
But,

1. **We can only store lexical instances. e.g. character strings.**
2. **They are lexical representations of entity instances.**

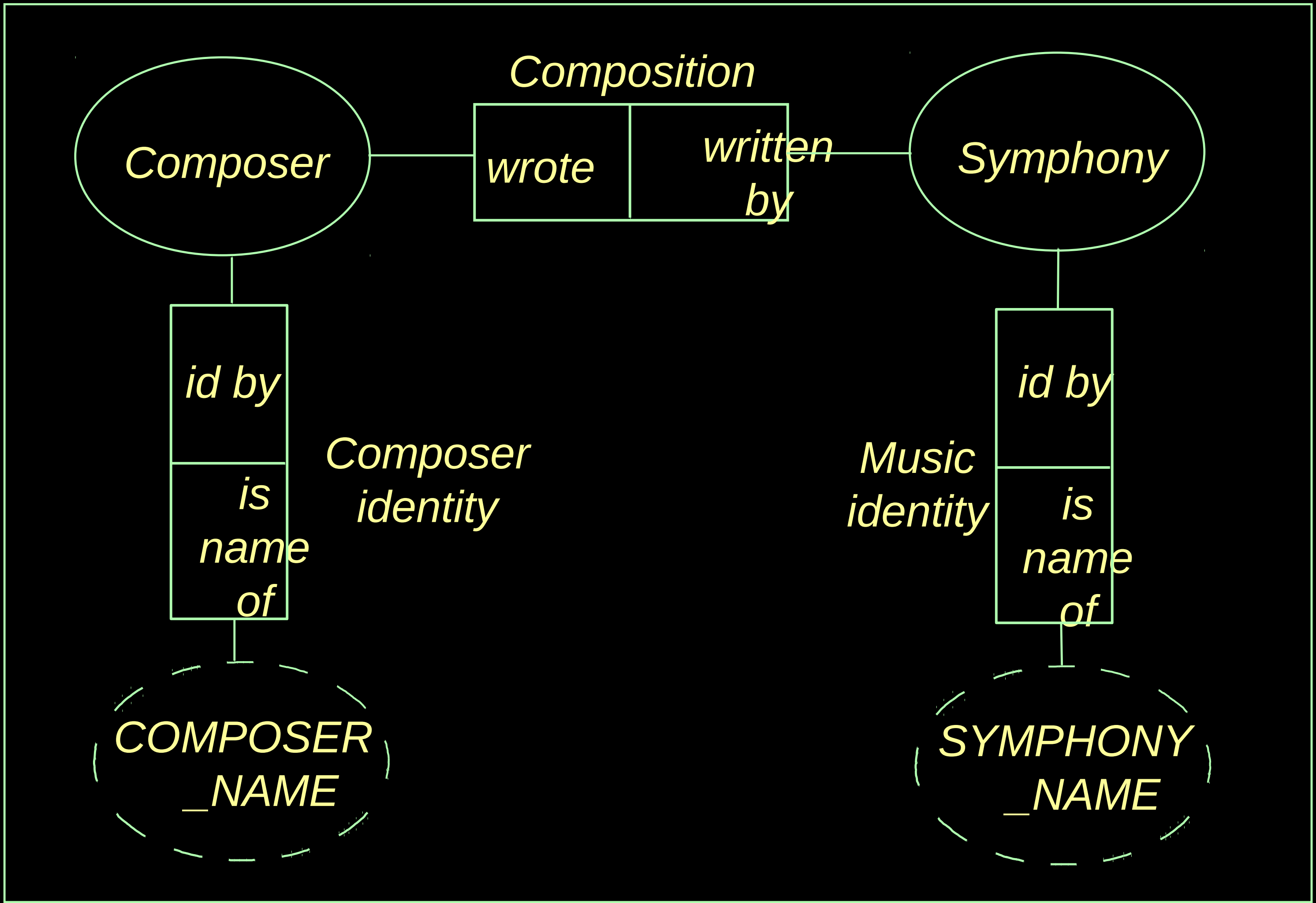
A very important message:

The most important thing to thoroughly understand when modelling information is the roles (and hence FACT types)

Example 2: Employee is identified by an EMPLOYEE_NUMBER

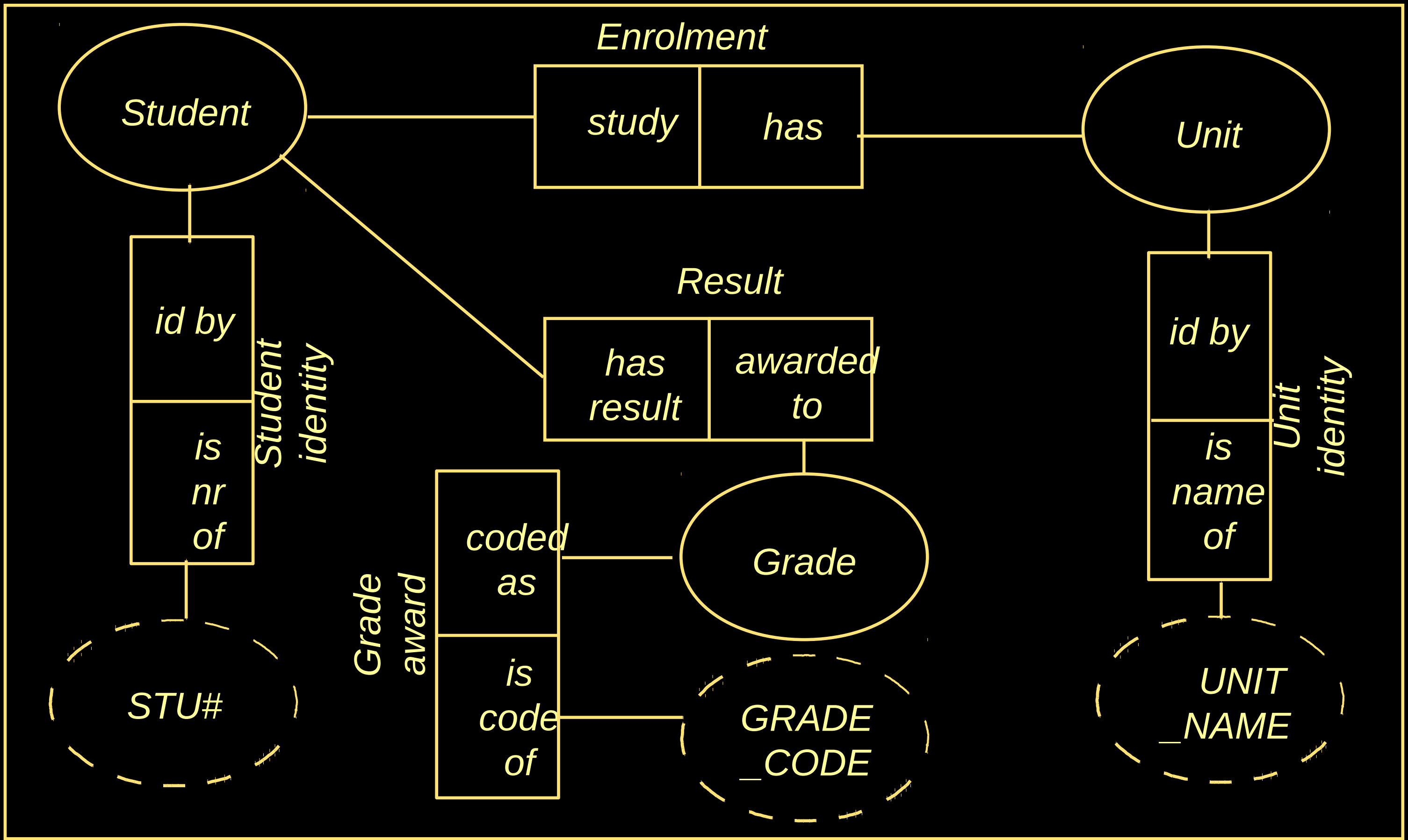


Example 3: Beethoven wrote the Pastoral Symphony



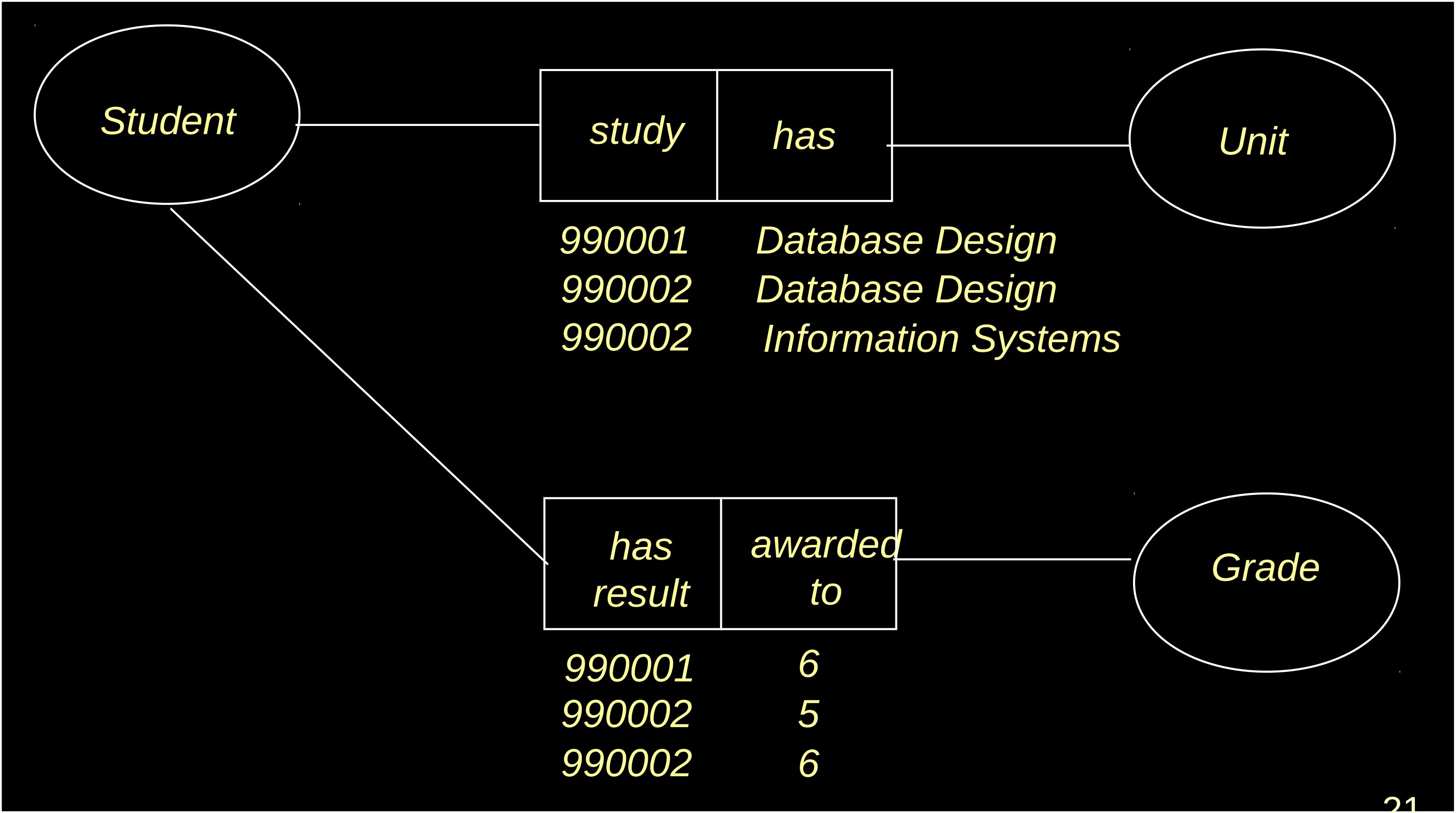
Example 4: Student 990001 has result of grade 6 for Database Design.

Would the following be correct? Is there any information lose?



Consider a significant population of the original sentence: And populate the facts

Student	Unit	Grade
990001	Database Design	6
990002	Database Design	5
990002	Information Systems	6



990001	Database Design
990002	Database Design
990002	Information Systems

990001	6
990002	5
990002	6

990001	Database Design
990002	Database Design
990002	Information Systems

$\bowtie$

990001	6
990002	5
990002	6

Carry out a natural join on the facts to see if the original population is recovered:

=

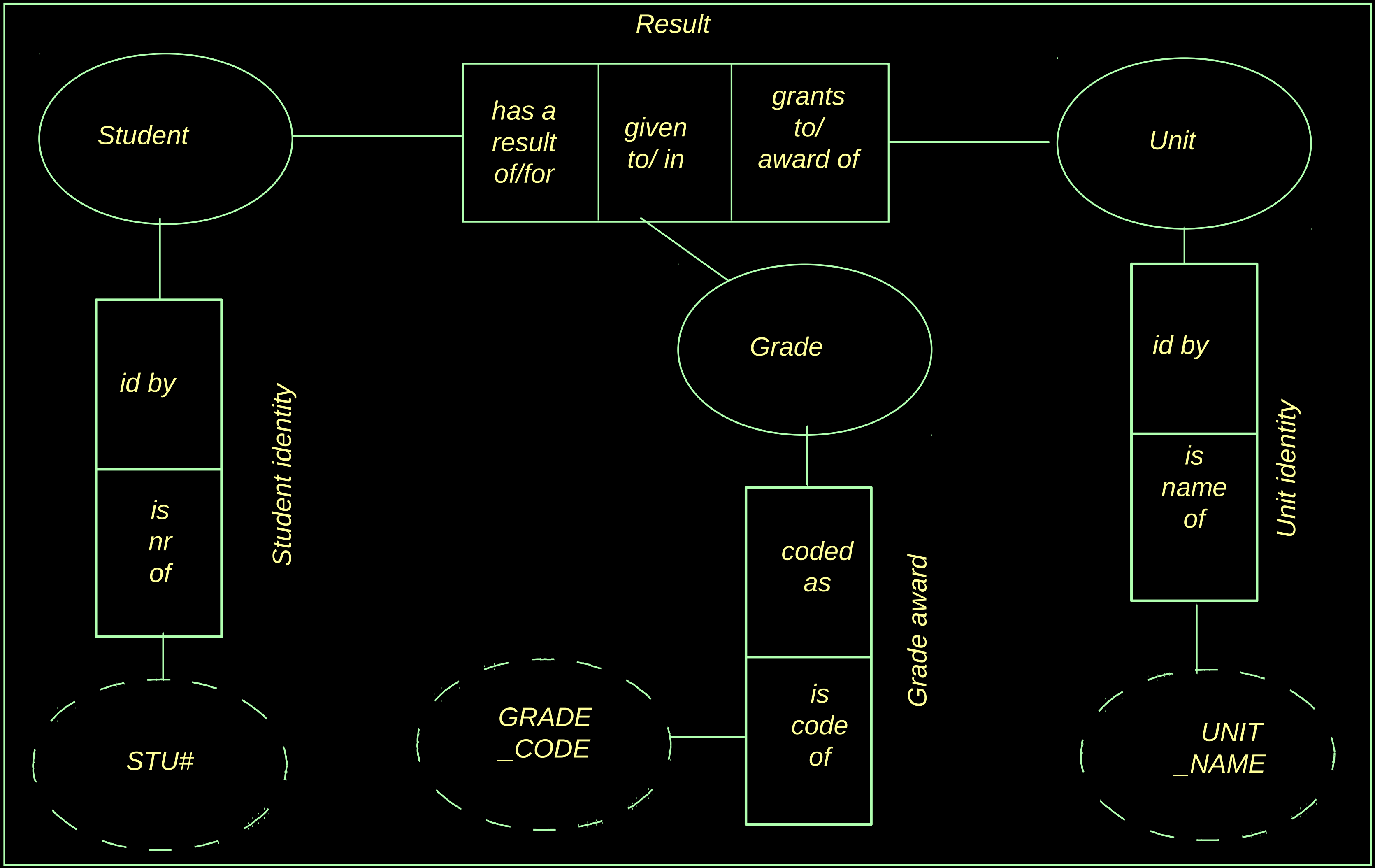
990001	Database Design	6
990002	Database Design	5
990002	Database Design	6
990002	Information Systems	5
990002	Information Systems	6

≠

Student	Unit	Grade
990001	Database Design	6
990002	Database Design	5
990002	Information Systems	6

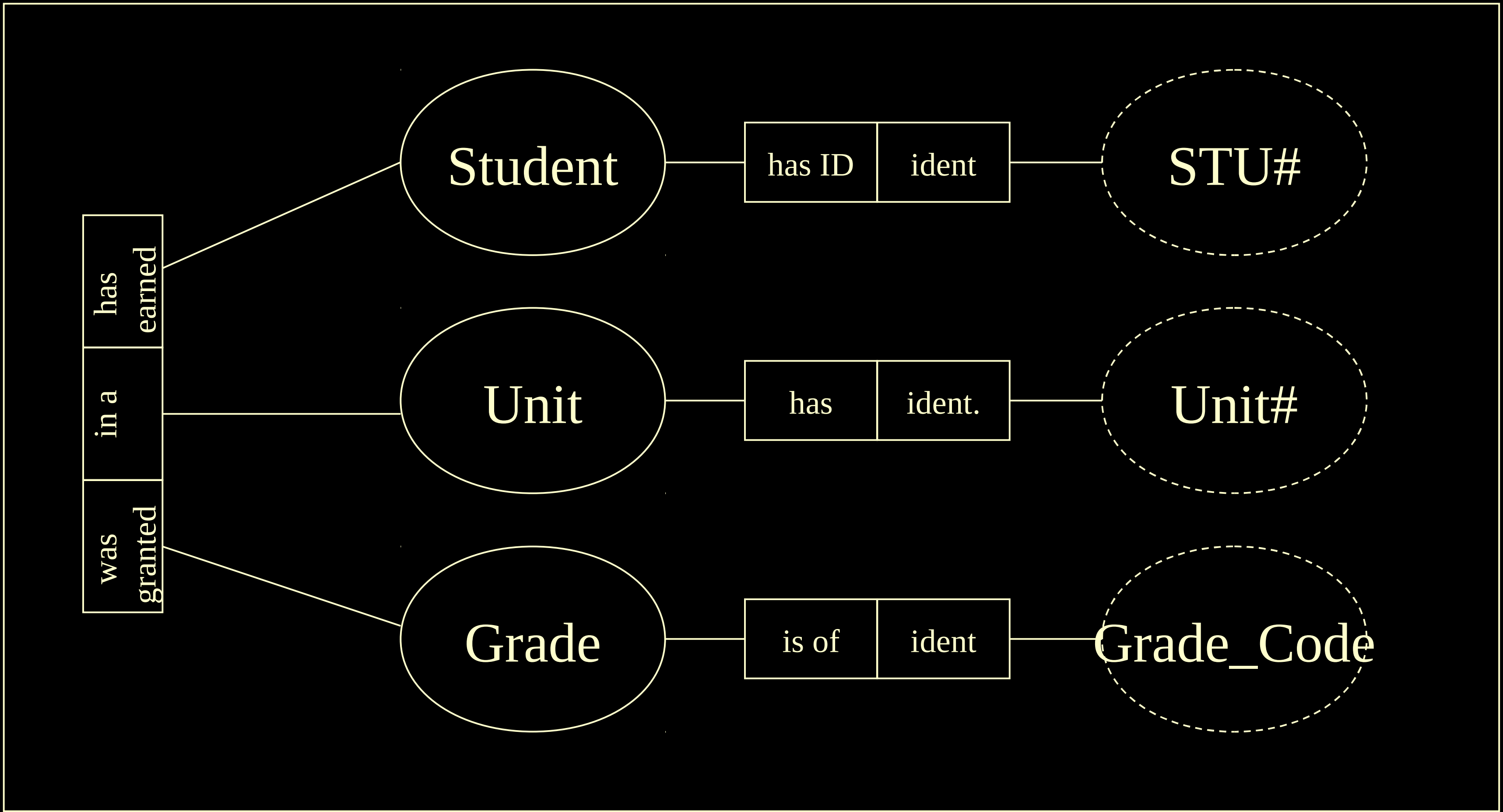
This result is incorrect. Therefore, the conceptual schema with two fact types is wrong.

Example 4: Student 990001 has result of grade 6 for Database Design.



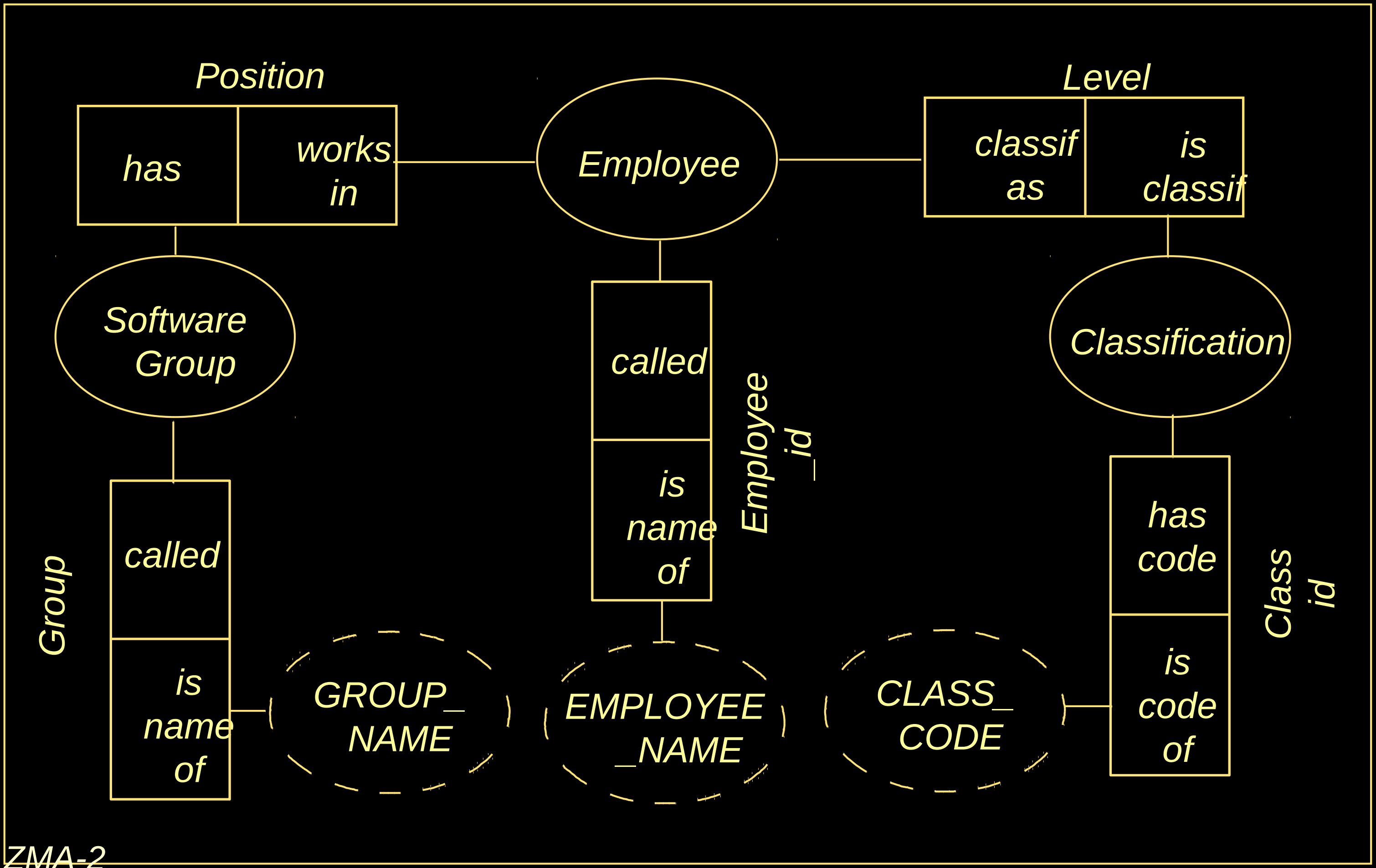
Notice the interdependency of the three roles.

Example 4: Student 990001 has result of grade 6 for Database Design.

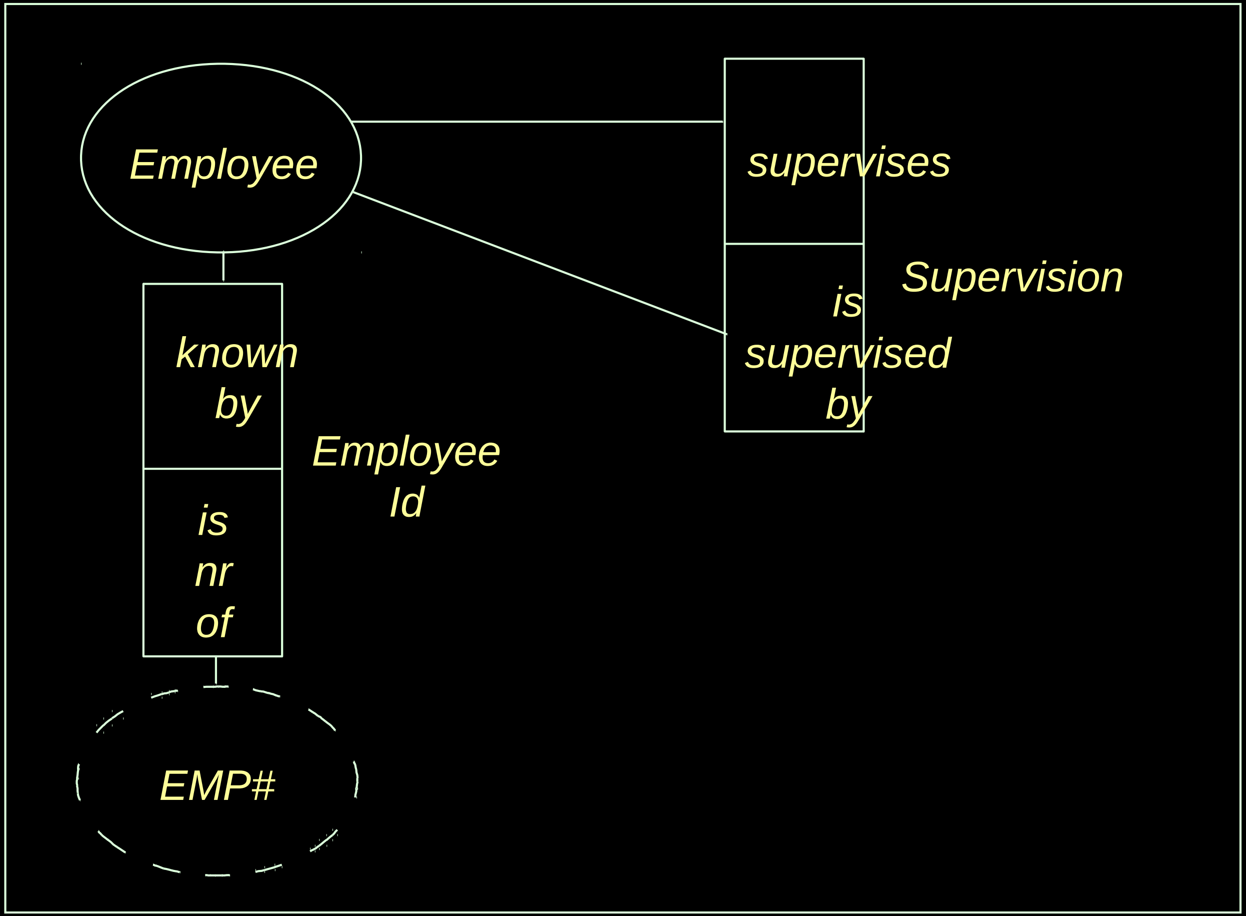


Example 5. David Brown works in Software Maintenance and has id CS01

The classification of the employee has been modelled as independent of the software group.



Example 6. Employee 143 is supervised by Employee 234



How to begin Conceptual Schema design?

- **It is usual to start from a sample output report which contains a significant population: the relationships between entity types can be inferred.**

Employee	*Department*	*Address*
E1	*D1*	*1 High St*
E2	*D2*	*15 Main Rd*
E3	*D1*	*1 High St*
E4	*D1*	*23 Toby St*
E5	*D2*	*5 Young St*
E6	*D3*	*44 High St*

There are many possible relationships (1:1, 1:n, n:m) that could be present in any output report.

Selected fact types:

The Employee with EMP# <...>
works_in
The Department with DEPT# <...>.

The Employee with EMP# <...>
lives_at
The Address with ADDR <...>.

The Department with DEPT# <...>
is located_at
The Address with ADDR <...>

The Employee with EMP# <...>
works in
The Department with DEPT# <...>
lives at
The Address with ADDR# <...>.

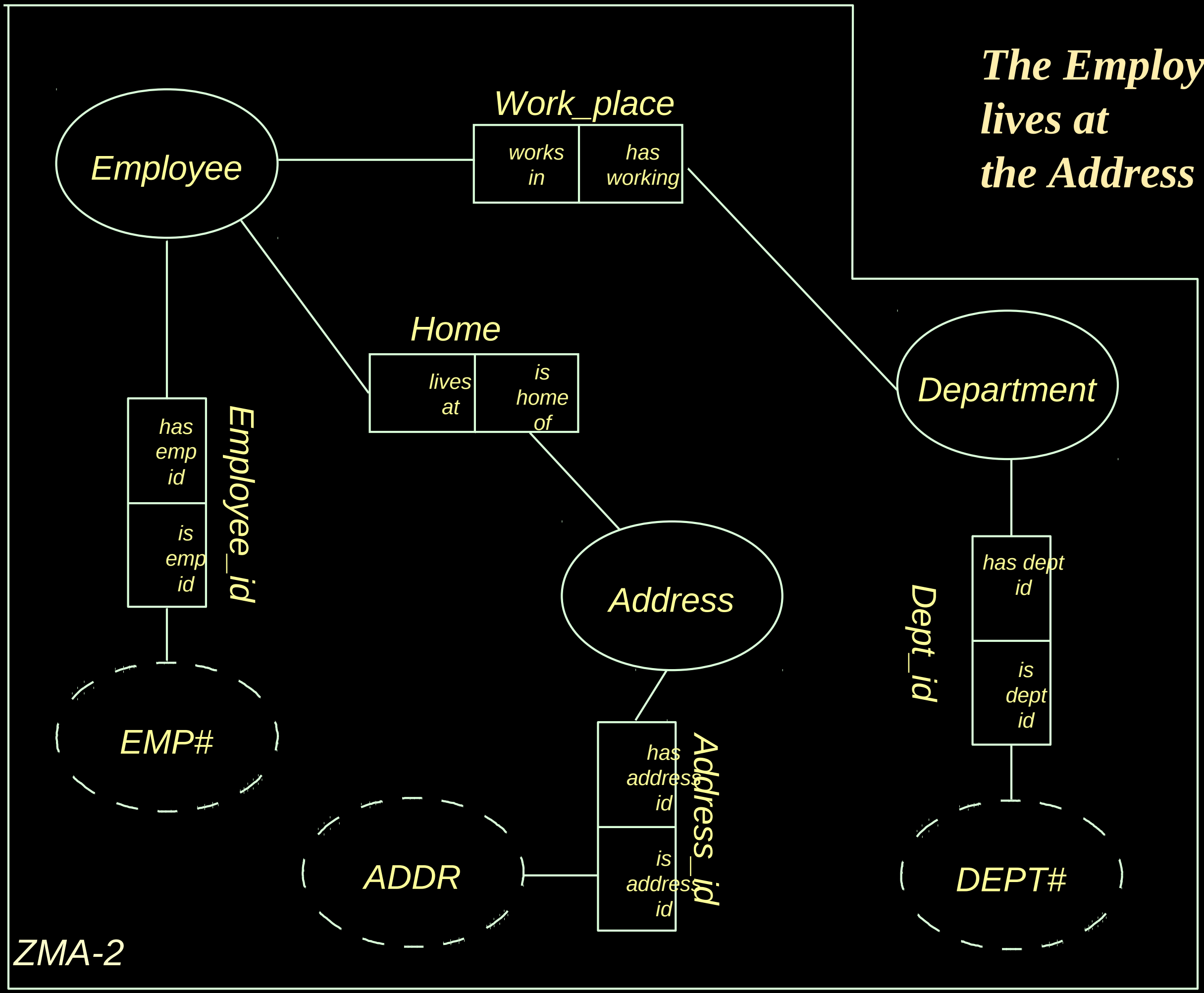
The skill the database designer has to exhibit is that of selecting which fact types capture meanings consistent with a Universe of Discourse out of all the possible fact types existing within an output report

Confirmation required from the user

From this output report it would be reasonable to select:

The Employee with EMP# <E1>
works in
the Department with DEPT# <D2>

The Employee with EMP# <E1>
lives at
the Address with ADDR <1 High St>



Constraints defined on Fact Types

Identificationtion of uniqueness constraints

1:1

1:n

n:m

Identification of entity type constraints

simple case

complex cases

Only binary fact types are considered at this stage

Identificationtion of uniqueness constraints

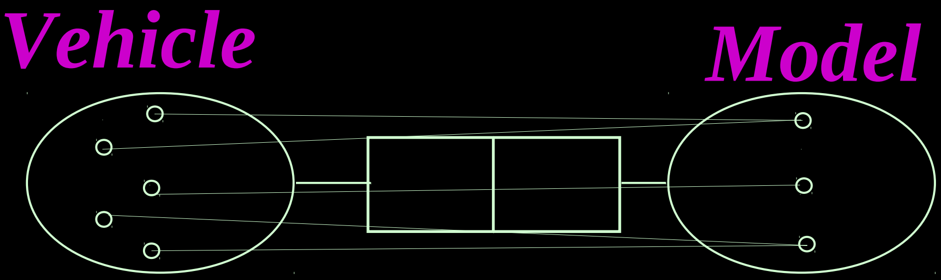
Vehicle	*Model*
REG#	*MODEL_NAME*
is_type	*has_reg_nr*
100 ABC *456 PQR* *345 ABC* *550 YUT* *987 WER*	*Camry* *Falcon* *Falcon* *Laser* *Camry*

In English, fact instances are:

(i) Vehicle 100 ABC is type Camry

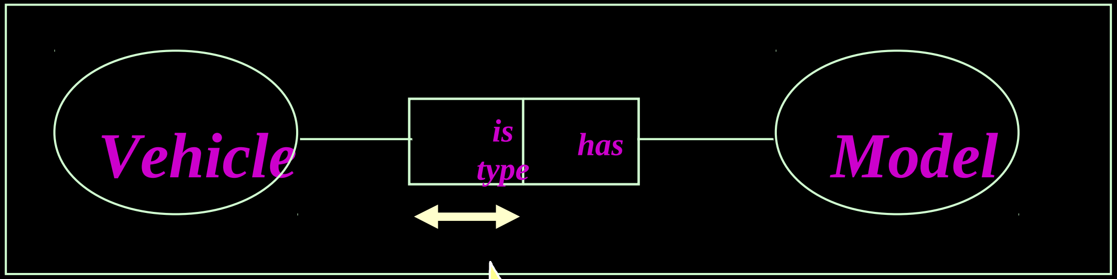
(ii) Vehicle 456 PQR is type Falcon

etc



1: N

Conceptual schema



There are two ways of viewing this fact type:

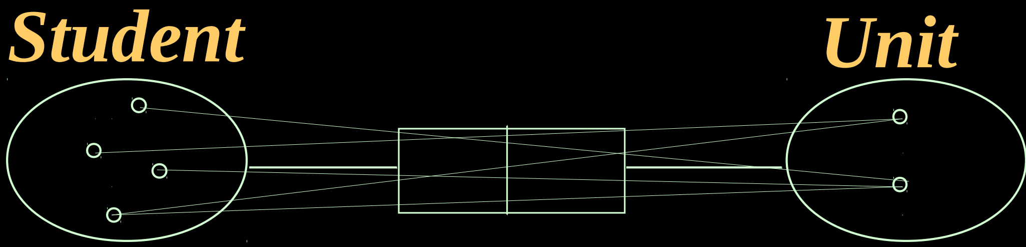
A Vehicle can be of only one Model

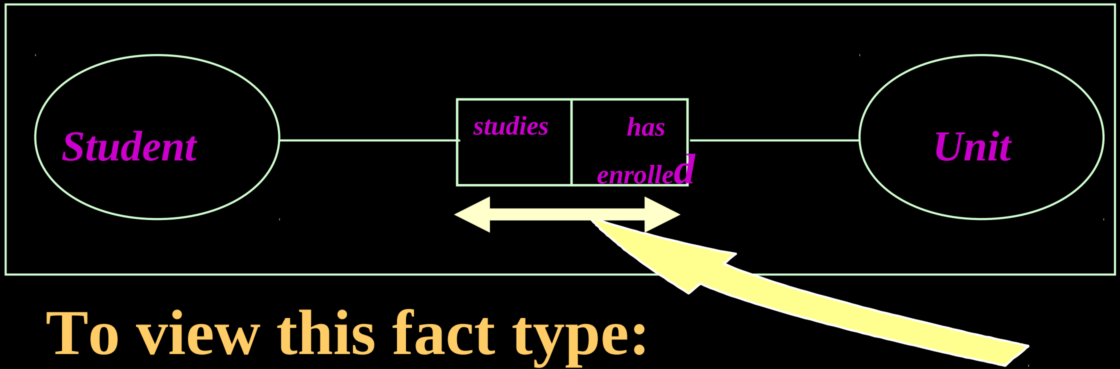
A Model can have many Vehicles

Representation of a ***uniqueness constraint***

Student	Unit
STU#	UNIT_CODE
studies	has_enrolled
990001	ITB100
990001	ITB200
990002	ITB100
990003	ITB200
990004	ITB100

N : M





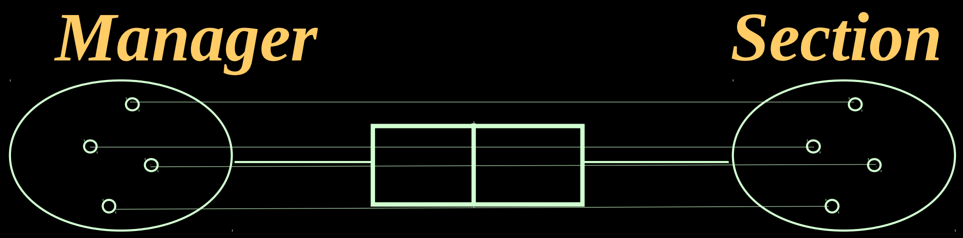
a uniqueness constraint

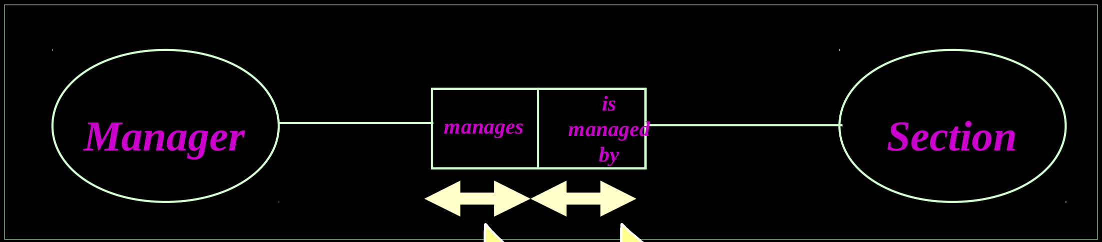
To view this fact type:

1. One student studies many Units and
   One Unit has enrollment of many Students
2. n Students study m Units

1 : 1

Manager	Section
EMP#	SECTION_CODE
manages	is_managed_by
13580	C3
87645	A4
12543	B2
45367	C4





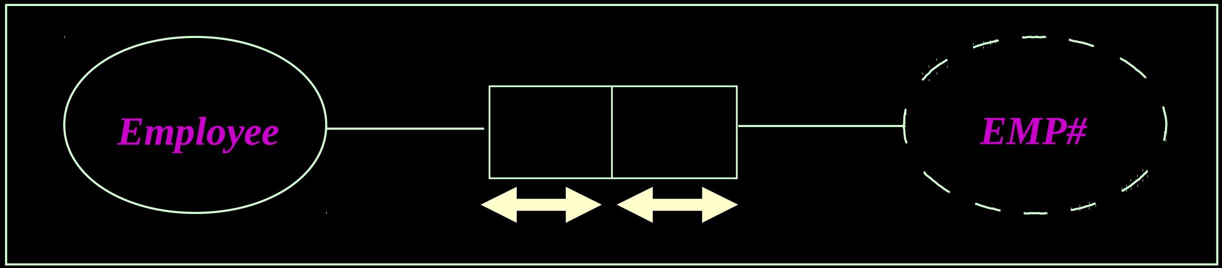
a uniqueness constraint

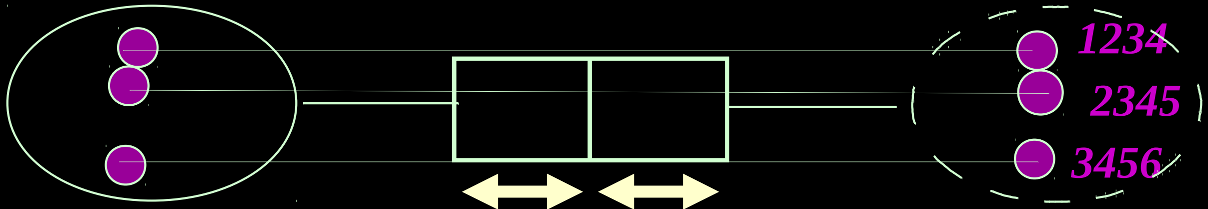
To view this:

1. One Manager manages one Section and
   One Section is_managed_by one Manager
2. 1 Manager manages 1 Section

Identification of Entity Types Instances

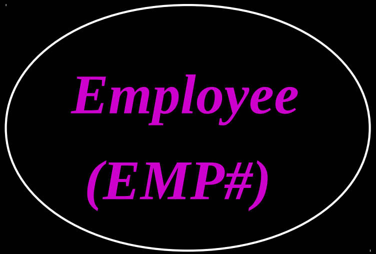
1. Simple case





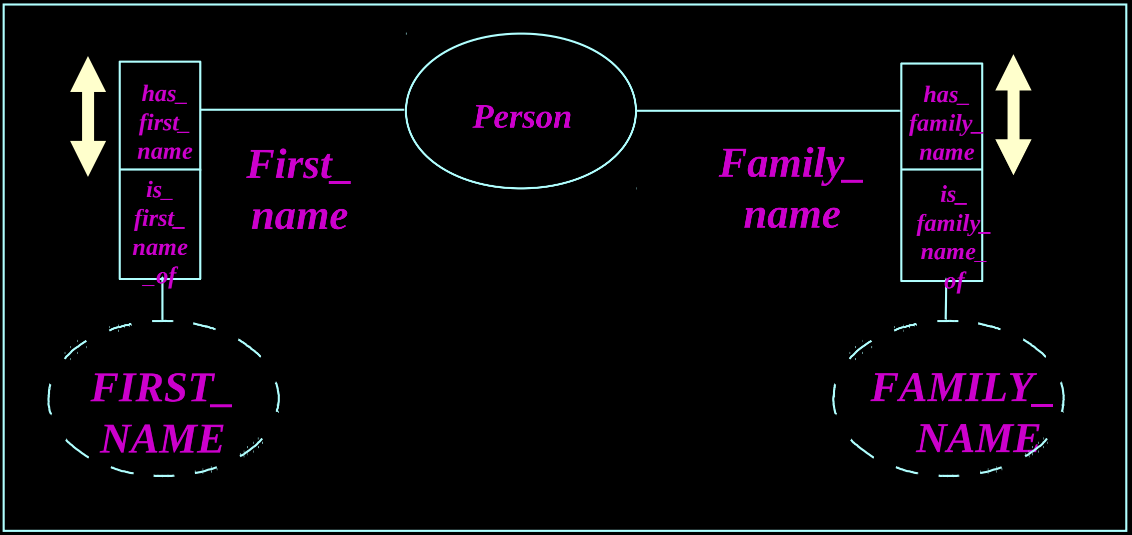
For each Employee there is one and only one EMP#. Similarly, for each EMP# there is one and only one Employee.

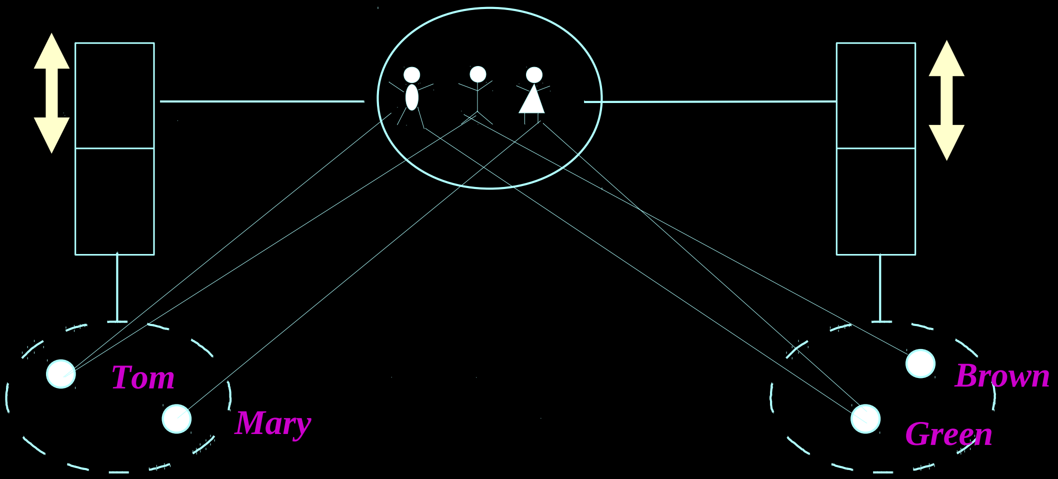
Abbreviated notation:

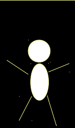
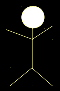


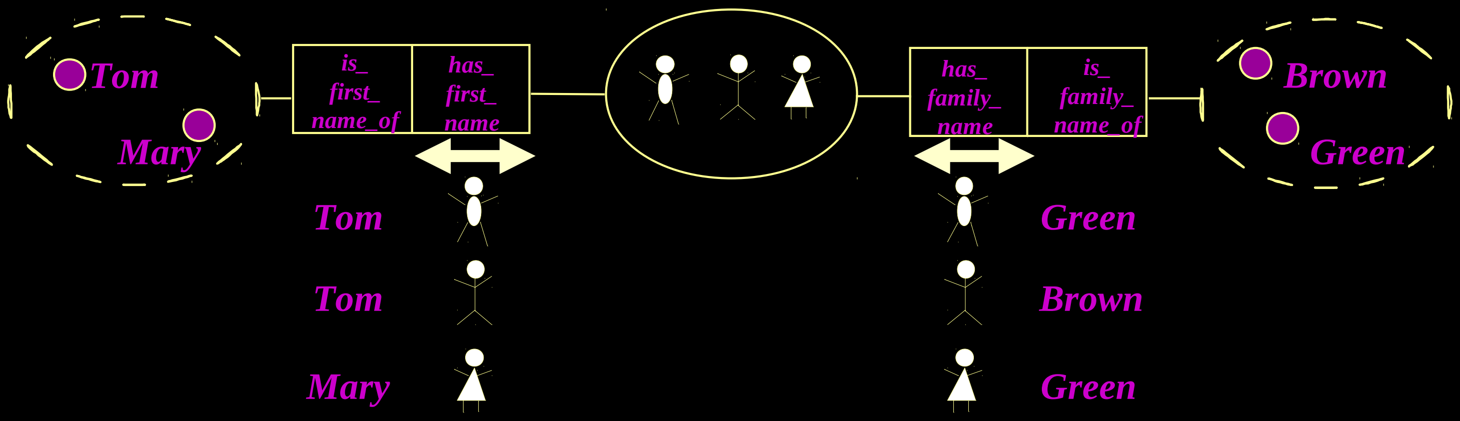
Note: such shortcut can be done only if there is 1:1 correspondence from entity type to label type, and from label type to entity type.

2. Complex case

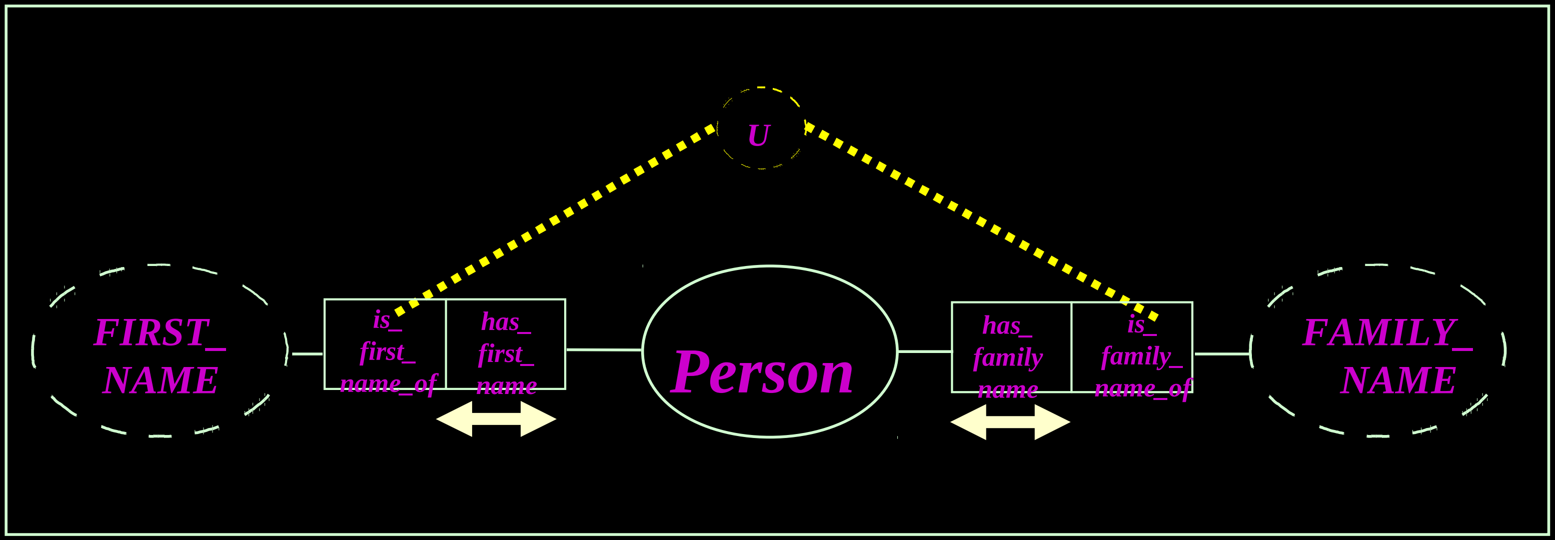




Person	first_ name	family_ name
	Tom	Green
	Tom	Brown
	Mary	Green



Inter Fact Type Uniqueness Constraint



The assumptions .....???

1. A Person has only one Family name, but a family name can be shared by many people.
2. A Person has only one first name, but it can be shared between many people.
3. However, the combination, of Family name and First name provides unique identification for a Person.

The last assumption is not true in general.

In the ORM conceptual schema, we are modeling (drawing) the entity types, label types and fact types, but not their instances!

However, it is very important for a designer to be able to populate the fact instances of fact types to verify the correctness of a schema,

ORM : Seven Steps of the Conceptual Schema Design Procedure (CSDP)

1. **Transform information examples into elementary facts, and apply quality checks.**
2. **Draw fact types, and apply population checks.**
3. **Check for arithmetic derivations of fact types, and superfluous entity types.**
4. **Add uniqueness constraints, and check the arity of fact types.**
5. **Add mandatory role constraints, and check logical derivations.**
6. **Add value, set comparisons, and subtyping constraints.**
7. **Add other constraints and perform final checks.**

Summary

- This lecture introduced informally the main concept and ORM philosophy.
- The precise way how to design correct conceptual schema will be covered on the next lecture.