SOFTWARE ENGINEERING LAB

Dr:- shimaa saad Assistant lecture,

Department of computer science

Email : shimaa_saad@yahoo.com

Telephone : 01222956341

EXCERCISE NO. 3

AIM :- To draw a sample

ENTITY RELATIONSHIP DIAGRAM

diagram for real project or system.

Hardware Requirements:

- Pentium 4 processor (2.4 GHz), 128 Mb RAM, Standard keyboard n mouse, colored monitor.

Software Requirements:

- Visio, Rational Rose, Windows XP,

AN ENTITY RELATIONSHIP DIAGRAM METHODOLOGY: (One way of doing it)

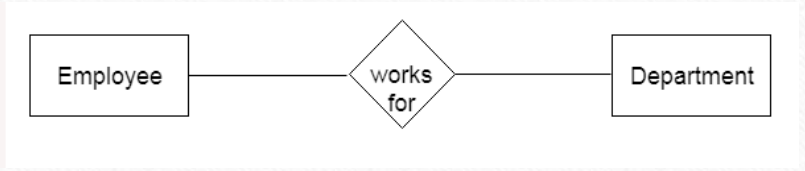
1. Identify Entities	Identify the roles, events, locations, tangible things or concepts about which the end-users want to store data.
2. Find Relationships	Find the natural associations between pairs of entities using a relationship matrix.
3. Draw Rough ERD	Put entities in rectangles and relationships on line segments connecting the entities.
4. Fill in Cardinality	Determine the number of occurrences of one entity for a single occurrence of the related entity.
5. Define Primary Keys	Identify the data attribute(s) that uniquely identify one and only one occurrence of each entity.
6. Draw Key-Based ERD	Eliminate Many-to-Many relationships and include primary and foreign keys in each entity.
7. Identify Attributes	Name the information details (fields) which are essential to the system under development.
8. Map Attributes	For each attribute, match it with exactly one entity that it describes.
9. Draw fully attributed ERD	Adjust the ERD from step 6 to account for entities or relationships discovered in step 8.
10. Check Results	Does the final Entity Relationship Diagram accurately depict the system data?

THEORY

- Entity Relationship Diagrams are a major data modelling tool and will help organize the data in your project into entities and define the relationships between the entities.
- This process has proved to enable the analyst to produce a good database structure so that the data can be stored and retrieved in a most efficient manner.

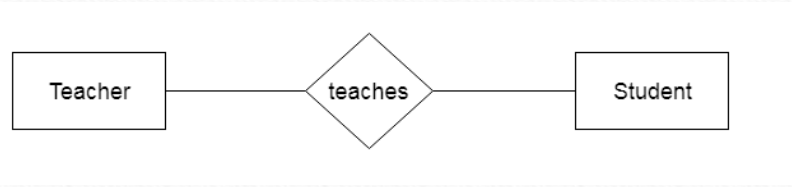
Entity

- A data entity is anything real or abstract about which we want to store data. Entity types fall into five classes: roles, events, locations, tangible things or concepts. E.g. employee, payment, campus, book.
- Specific examples of an entity are called instances.



- **An entity may be any object, class, person or place.**
- **In the ER diagram, an entity can be represented as rectangles.**
- 1. Identify Entities Identify the roles, events, locations, tangible things or concepts about which the end-users want to store data

2. Relationship



A data relationship is a natural association that exists between one or more entities. **A relationship is used to describe the relation between entities.**

- E.g. Employees process payments.

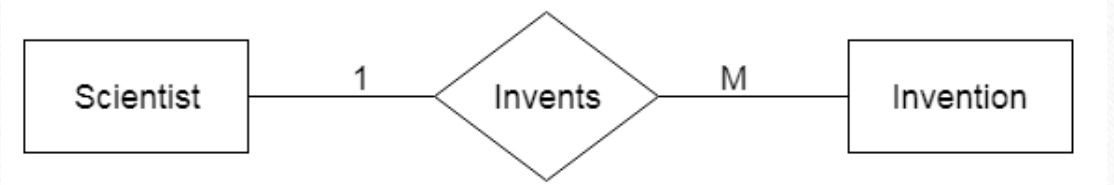
Cardinality defines the number of occurrences of one entity for a single occurrence of the related entity.

- E.g. an employee may process many payments but might not process any payments depending on the nature of her job

Types of relationship

- **One-to-One**

- **One-to-many**



- **Many-to-many**

ربط الجداول

- المقصود بربط الجداول هو إنشاء علاقة ارتباط دائمة بين جدولين أو أكثر , يكون من نتيجتهما استخراج بيانات من كلا الجدولين و إظهارهما في النماذج أو التقارير أو الأستعلام .

قبل الربط لابد من تأسيس علاقة ارتباط – Relationship

شروط إنشاء العلاقة بين جدولين :

- يجب أن يشتمل كلا الجدولين على حقل متماثل في البيانات .
- أن يشتمل أحد الجدولين على مفتاح رئيسي .
- أن يكون كلا الجدولين مخزن في نفس قاعدة البيانات .

علاقة ارتباط رأس بأطراف : One –To – Many

- تعني أن السجل الواحد في جدول البيانات الرئيسي Primary Table يقابله أكثر من سجل في جدول آخر يسمى الجدول المرتبط . Related Table
- هذا النوع من العلاقات هو الأكثر استخداما .
- كمثال , جدول الناشرين وجدول الكتب , لهم علاقه one-to-many حيث أن كل ناشر له العديد من الكتب , ولكن كل كتاب له ناشر واحد فقط

جدول الطالبات

الرقم الجامعي	الاسم	التخصص	المستوى	المكافأة	انتظام	تاريخ الميلاد	العنوان	رقم الهاتف
433001	منال احمد	رياضيات	1	1000	نعم	1/3/2010	القاهرة	12321
433002	مها محمد	فيزياء	2	1000	نعم	2/11/2009	القاهرة	12414
433003	اسماء خالد	رياضيات	1	1000	نعم	4/5/2010	القاهرة	12414

جدول الدرجات

الرقم الجامعي	المادة	المستوى	الدرجة
433003	دراسات	1	100
433003	عربي	1	99
433003	رياضه	1	95

Many – To – Many : علاقة اطراف باطراف

- تعني أن كل سجل من الجدول الرئيسي يقابله عدة سجلات في الجدول المرتبط ويقابل السجل الواحد في الجدول المرتبط عدة سجلات في الجدول الرئيسي , مثلا في الجامعة :

- يوجد جدول المواد و جدول الأساتذة في جدول المواد يوجد مادة يعطيها أكثر من أستاذ و في جدول الأساتذة يوجد أستاذ يعطي أكثر من مادة .

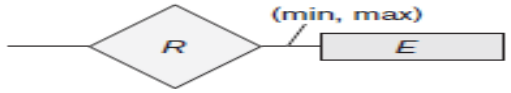
- يتطلب هذا النوع من العلاقات جدول ثالث يسمى جدول ربط .

One –To – One : علاقة ارتباط رأس برأس

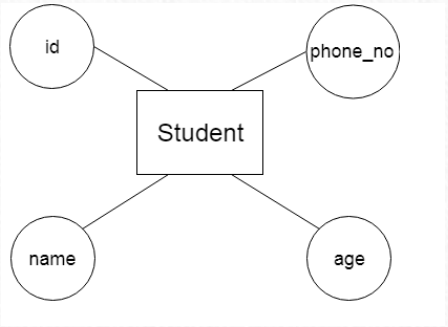
- تعني أن السجل الواحد في جدول البيانات الرئيسي Primary Table يقابله سجل واحد فقط في الجدول المرتبط به .

- غالبا يستخدم عندما نريد فصل معلومات السجل إلى بيانات عامة وبيانات خاصة ,
مثلا :
معلومات الموظف من اسم وعنوان ورقم هاتف في جدول أساسي والمعلومات الخاصة مثل الراتب في جدول آخرمرتبط .

- هذا النوع من العلاقات هو الأقل استخداما .

Symbol	Meaning
	Entity
	Weak Entity
	Relationship
	Indentifying Relationship
	Attribute
	Key Attribute
	Multivalued Attribute
	Composite Attribute
	Derived Attribute
	Total Participation of E_2 in R
	Cardinality Ratio 1: N for E_1:E_2 in R
	Structural Constraint (min, max) on Participation of E in R

Attribute



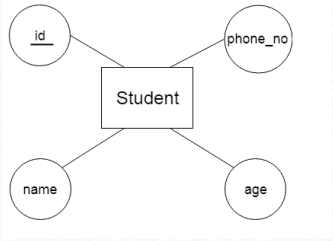
A data attribute is a characteristic common to all or most instances of a particular entity. Synonyms include property, data element, field.

The attribute is used to describe the property of an entity. Eclipse is used to represent an attribute.

For example, id, age, contact number, name, etc. can be attributes of a student.

An attribute or combination of attributes that uniquely identifies one and only one instance of an entity is called a **primary key** or identifier.

- E.g. Employee Number is a primary key for Employee.



- 3. Draw Rough ERD

Put entities in rectangles and relationships on line segments connecting the entities.

- 4. Fill in Cardinality

Determine the number of occurrences of one entity for a single occurrence of the related entity.

- 5. Define Primary Keys

Identify the data attribute(s) that uniquely identify one and only one occurrence of each entity.

- 6. Draw Key-Based ERD

Eliminate Many-to-Many relationships and include primary and foreign keys in each entity.

المفاتيح Keys

- المفتاح هو عبارة عن حقل أو أكثر أي عمود أو أكثر يتم اختياره من بين حقول الجدول وفقا لشروط معينة.
- أهم استخدام لحقل المفتاح هو تمييز السجلات عن بعضها (مثل المفتاح الرئيسي أو المركب)

مثال:

في جدول يحتوي بيانات الطلاب فإن أنسب مفتاح رئيسي للجدول هو "الرقم الجامعي" إذ انه :

- يحتوي قيم لا يمكن أن تتكرر.
- لا يوجد طالبان في الجامعة لهما نفس الرقم الجامعي.

فمثلا لا نستطيع تعيين الحقل "اسم الطالب" كمفتاح رئيسي لانه من الممكن تكرار البيانات فيه. لأن أسماء الناس تتكرر دائما.

انواع المفاتيح

الرقم المتسلسل	الاسم	العلامة
1	محمد	25
2	أمل	25
3	أكرم	15
4	أماني	13

أي من الحقول في الجدول السابق يصلح ليكون مفتاح رئيسي؟

الرقم المتسلسل

رقم السيارة	النوع	اللون	تاريخ الصنع
36728	مرسيدس	أسود	1999
65474	مرسيدس	أحمر	1999
74633	مرسيدس	أحمر	1999
67467	هوندا	أسود	2000

أي من الحقول في الجدول السابق يصلح ليكون مفتاح رئيسي؟

رقم السيارة

رقم الطالب	المادة المسجلة	القاعة
20009090	ن ح 101	م ق 102
20009090	ع ح 210	م ق 102
20009090	م م 210	أق 109
20018888	ن ح 101	م ق 102
20018888	م م 210	أق 107

أي من الحقول في الجدول السابق يصلح ليكون مفتاح للجدول؟

نلاحظ أن البيانات في جميع الحقول من الممكن أن تتكرر. لذلك نبحث في إمكانية إنشاء مفتاح مركب. لنجد أن الحقلين "رقم الطالب" و "المادة المسجلة" يشكلان مفتاح مركب.

- 7. Identify Attributes

Name the information details (fields) which are essential to the system under development.

- 8. Map Attributes

For each attribute, match it with exactly one entity that it describes.

- 9. Draw fully attributed ERD

Adjust the ERD from step 6 to account for entities or relationships discovered in step 8.

- 10.Check Results

Does the final Entity Relationship Diagram accurately depict the system data?

A SIMPLE EXAMPLE

- A company has several departments.
- Each department has a supervisor and at least one employee.
- Employees must be assigned to at least one, but possibly more departments. At least one employee is assigned to a project, but an employee may be on vacation and not assigned to any projects.
- The important data fields are the names of the departments, projects, supervisors and employees, as well as the supervisor and employee number and a unique project number.

1. Identify Entities

- The entities in this system are Department, Employee, Supervisor and Project. One is tempted to make Company an entity, but it is a false entity because it has only one instance in this problem. True entities must have more than one instance.
- E.g. a student registration form would refer to Student (a role), but also Course (an event), Instructor (a role), Advisor (a role), Room (a location), etc.

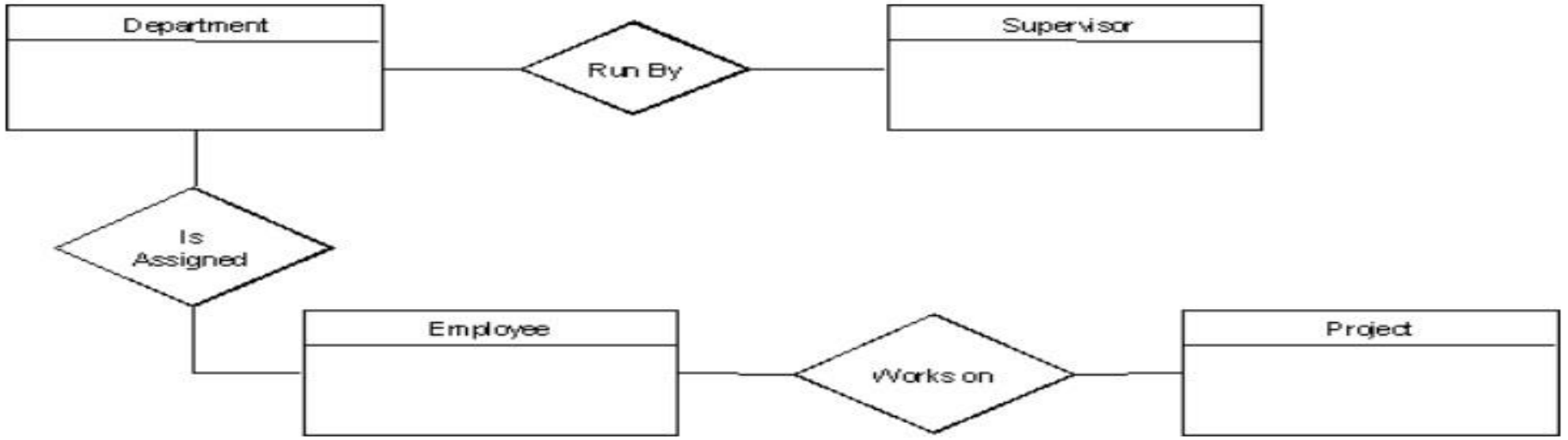
2. Find Relationships

We construct the following Entity Relationship Matrix:

	Department	**Employee**	**Supervisor**	**Project**
Department		is assigned	run by	
Employee	belongs to			works on
Supervisor	runs			
Project		uses		

3. Draw Rough ERD

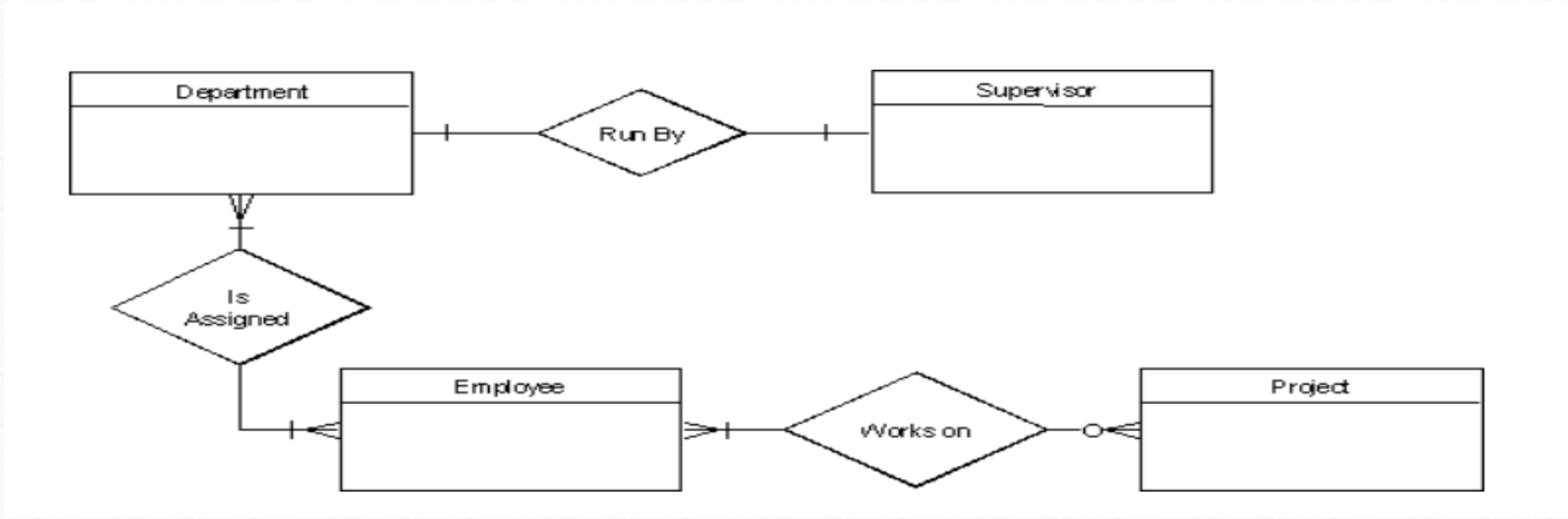
We connect the entities whenever a relationship is shown in the entity Relationship Matrix.



- Using rectangles for entities and lines for relationships, we can draw an Entity Relationship Diagram (ERD).

4. Fill in Cardinality

- From the description of the problem we see that:
- Each department has exactly one supervisor.
- A supervisor is in charge of one and only one department.
- Each department is assigned at least one employee.
- Each employee works for at least one department.
- Each project has at least one employee working on it.
- An employee is assigned to 0 or more projects.
- At each end of each connector joining rectangles, we need to place a symbol indicating the minimum and maximum number of instances of the adjacent rectangle there are for one instance of the rectangle at the other end of the relationship line.


Department
Run By
Supervisor
Is
Assigned
Employee
Works on
Project

5. Define Primary Keys

- The primary keys are Department Name, Supervisor Number, Employee Number, Project Number.
- For each entity we must find a unique primary key so that instances of that entity can be distinguished from one another

6. Draw Key-Based ERD

- There are two many-to-many relationships in the rough ERD above, between Department and Employee and between Employee and Project. Thus we need the associative entities Department-Employee and Employee-Project. The primary key for Department-Employee is the concatenated key Department Name and Employee Number. The primary key for Employee Project is the concatenated key Employee Number and Project Number.
- It will have a 1-1 relationship
- The key-based ERD has no many-to-many relationships and each entity has its primary and foreign keys listed below the entity name in its rectangle.

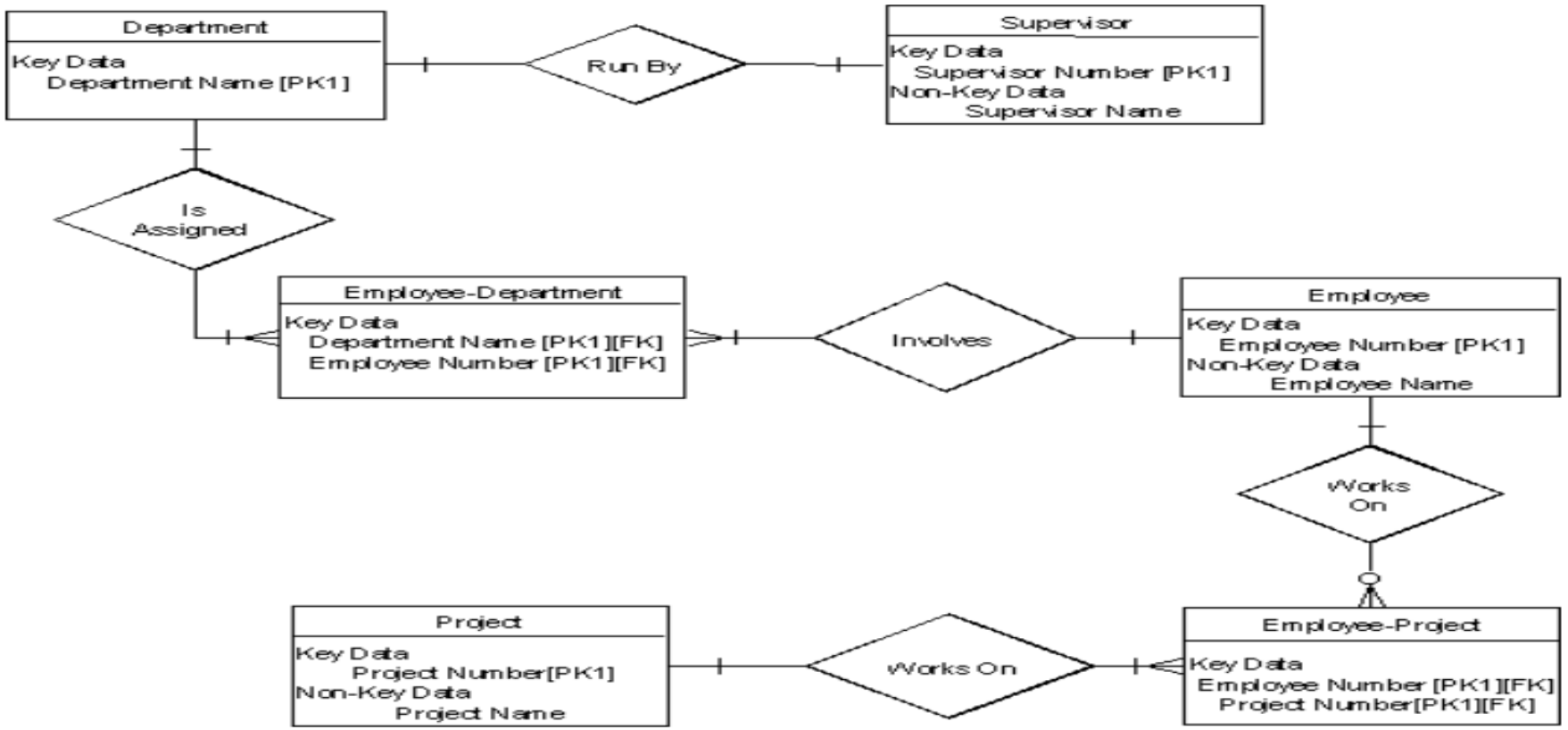
- **7. Identify Attributes**
- The only attributes indicated are the names of the departments, projects, supervisors and employees, as well as the supervisor and employee NUMBER and a unique project number.

- In this step we try to identify and name all the attributes essential to the system we are studying without trying to match them to particular entities.
- **8. Map Attributes**
- For each attribute we need to match it with exactly one entity.

8. Map Attributes

Attribute	Entity	Attribute	Entity
Department Name	Department	Supervisor Number	Supervisor
Employee Number	Employee	Supervisor Name	Supervisor
Employee Name	Employee	Project Name	Project
		Project Number	Project

9. Draw Fully Attributed ERD



10. Check Results

- The final ERD appears to model the data in this system well.
- Look at your diagram from the point of view of a system owner or user. Is everything clear? Check through the Cardinality pairs. Also, look over the list of attributes associated with each entity to see if anything has been omitted.

EXCERCISE NO. 4

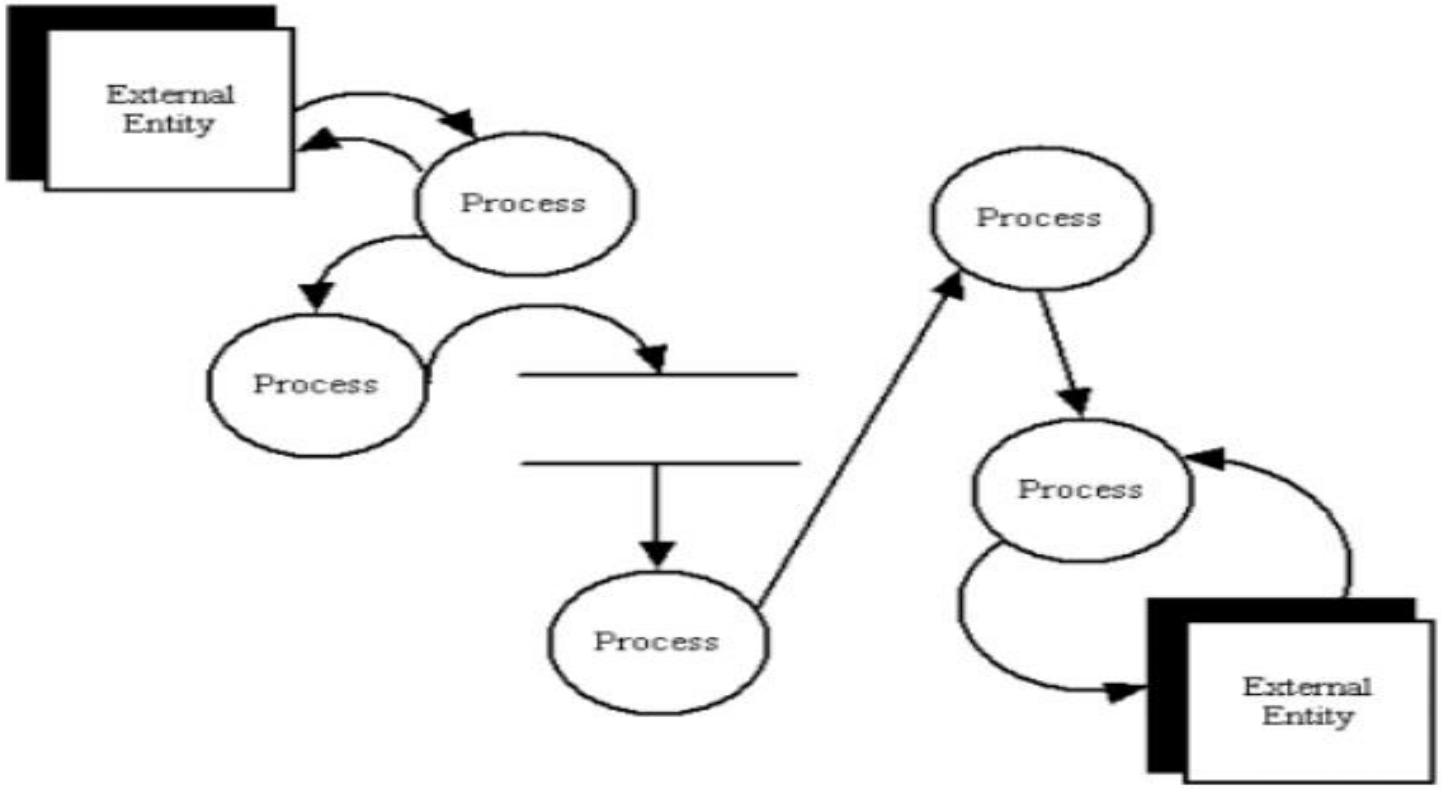
- AIM: To prepare DATA FLOW DIAGRAM for any project.

REQUIREMENTS:

- **Hardware Interfaces**
- ▪ Pentium(R) 4 CPU 2.26 GHz, 128 MB RAM
- ▪ Screen resolution of at least 800 x 600 required for proper and complete viewing of screens. Higher resolution would not be a problem.
- ▪ CD ROM Driver
- **Software Interfaces**
- ▪ Any window-based operating system (Windows 95/98/2000/XP/NT)
- ▪ WordPad or Microsoft Word

THEORY

Data flow diagrams illustrate how data is processed by a system in terms of inputs and outputs.

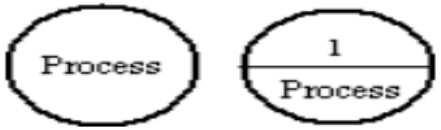


Data Flow Diagram Notations

Data Flow Diagram Notations

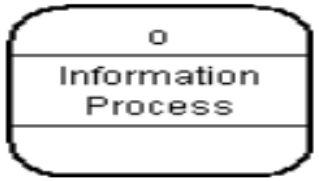
You can use two different types of notations on your data flow diagrams: Yourdon & Coad or Gane & Sarson.

Process Notations



Yourdon and Coad
Process Notations

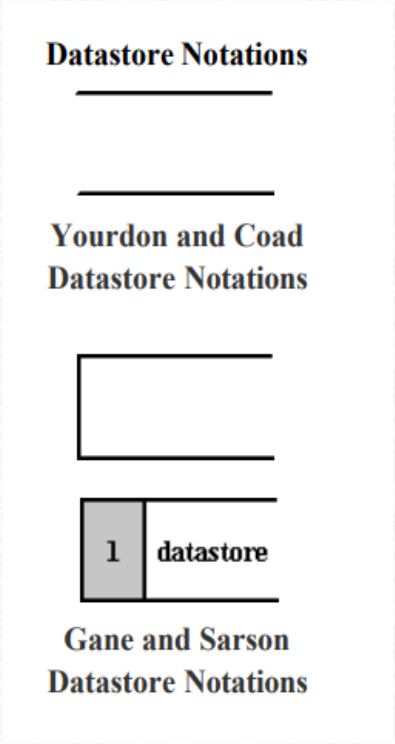
Process



Gane and Sarson
Process Notation

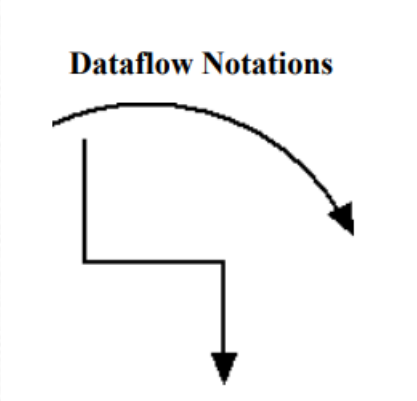
A process transforms incoming data flow into outgoing data flow.

A process transforms incoming data flow into outgoing data flow.



DataStore

- Datastores are repositories of data in the system. They are sometimes also referred to as files.



Dataflow

Dataflows are pipelines through which packets of information flow.

Label the arrows with the name of the data that moves through it

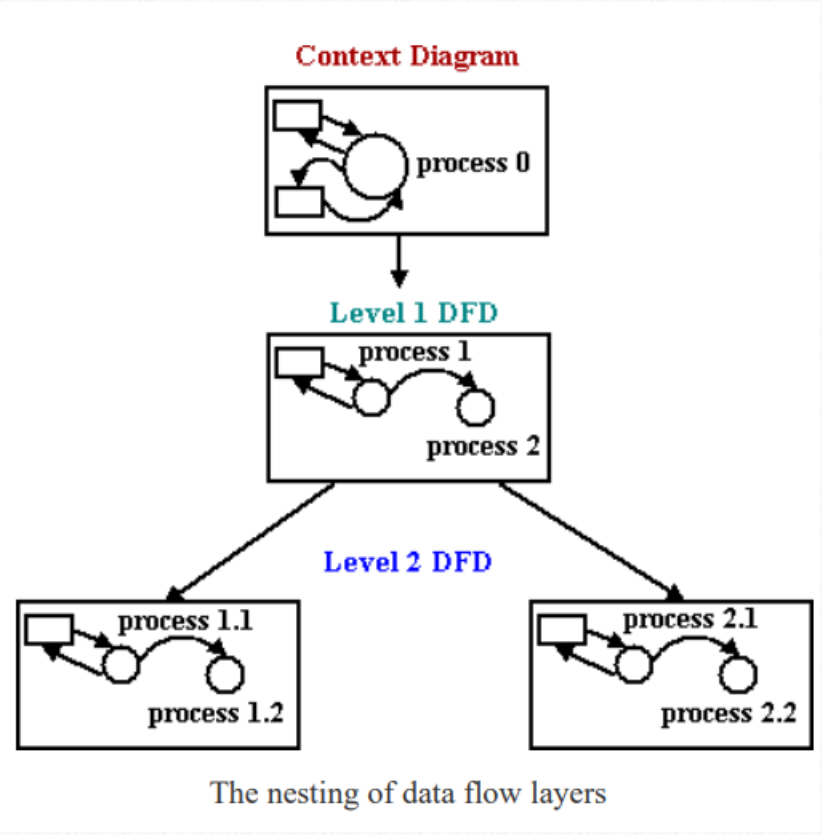
HOW TO DRAW DATA FLOW DIAGRAMS (cont'd)

- Data Flow Diagram Layers

Draw data flow diagrams in several nested layers.

A single process node on a high level diagram can be expanded to show a more detailed data flow diagram.

Draw the context diagram first, followed by various layers of data flow diagrams.

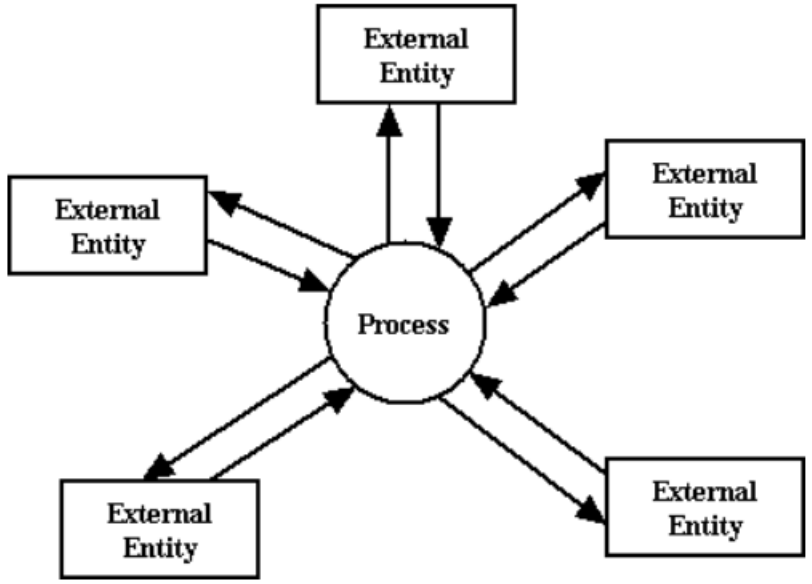


The nesting of data flow layers

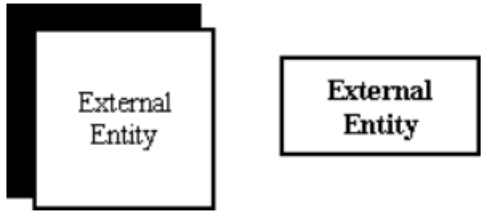
Context Diagrams

A context diagram is a top level (also known as Level 0) data flow diagram.

It only contains one process node (process 0) that generalizes the function of the entire system in relationship to external entities.



External Entity Notations

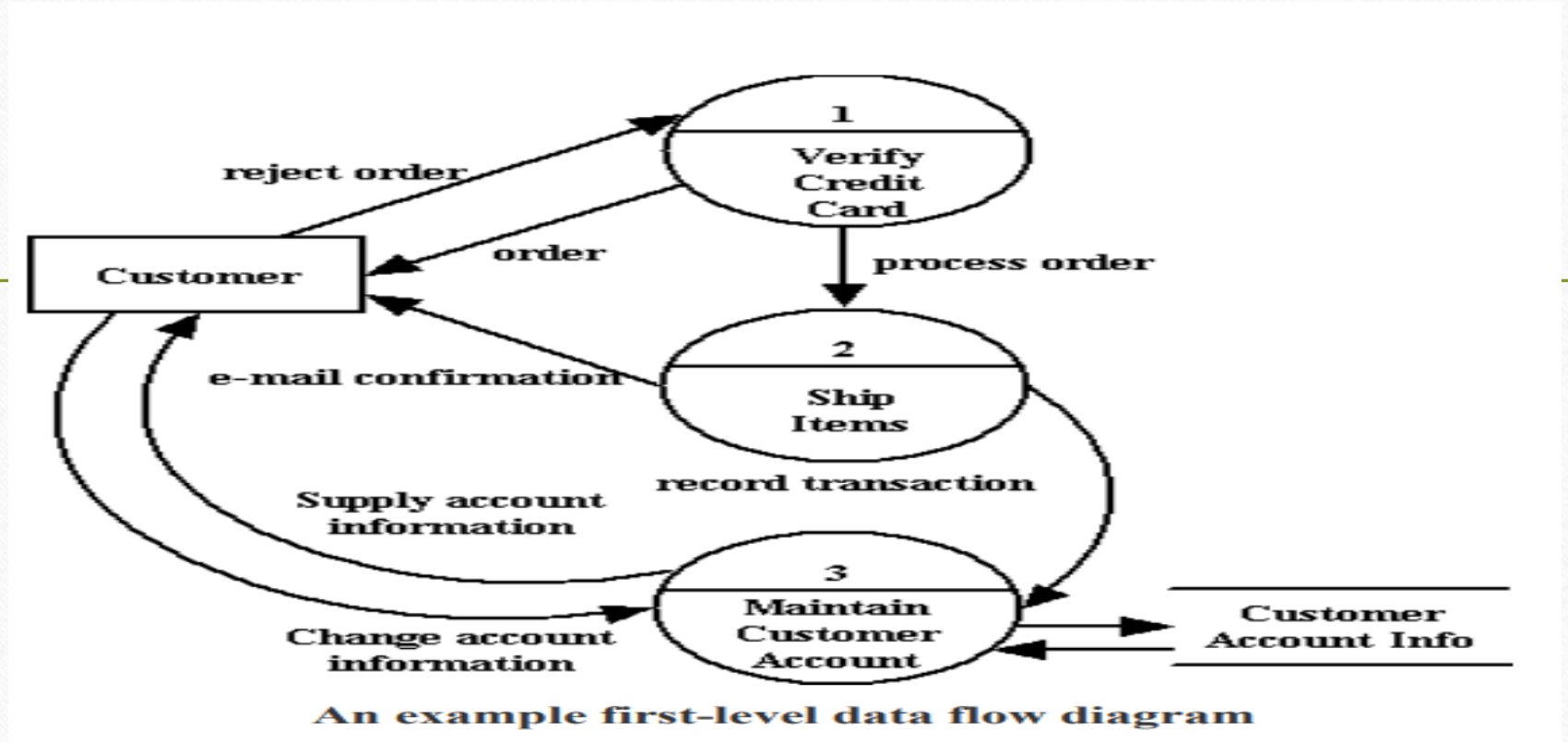


- External Entity

External entities are objects outside the system, with which the system communicates. External entities are sources and destinations of the system's inputs and outputs.

- DFD levels

The first level DFD shows the main processes within the system. Each of these processes can be broken into further processes until you reach pseudocode.



An example first-level data flow diagram

- Conclusion: The dataflow diagram was made successfully by following the steps described above.