Lecture Notes # 1

It is the aim of this script to give students basic concepts and applications of databases. Students must be aware of advantages in data management, especially for statistical analysis, by means of databases. This script provides language and method to translate "stakeholder" requirements - specifications - into a database (conceptual view) [1].

Introduction to Databases

Databases today are essential to every business.

> "Whenever you visit a major web site: Google, Wikipedia, Amazon, Booking.com, even services offered by the Pubic Administration (PA) or thousands of different other sites (including your preferred social: Instagram, TikTok, Facebook, Twitter, Youtube, and your phone calls, chats), there is a database behind, serving up the information you request.
>
> Corporations maintain all their important records in databases for its business.
>
> Databases are likewise found at the core of many scientific investigations. They represent the data gathered by astronomers, by investigators of the human genome, biochemists exploring properties of proteins, health medical records, along with many other scientist activities."[1]

What is a database? A database is nothing more than a collection of information that exists over a long period of time, often many years. Commonly the term **database** refers to a **collection** of data which is managed by a **Database Management System (DBMS)**.

[1] The script is mainly based on "A First Course in Database Systems", J. Ullman, J. Widom. Mostly examples and exercises are created by the author of this script.

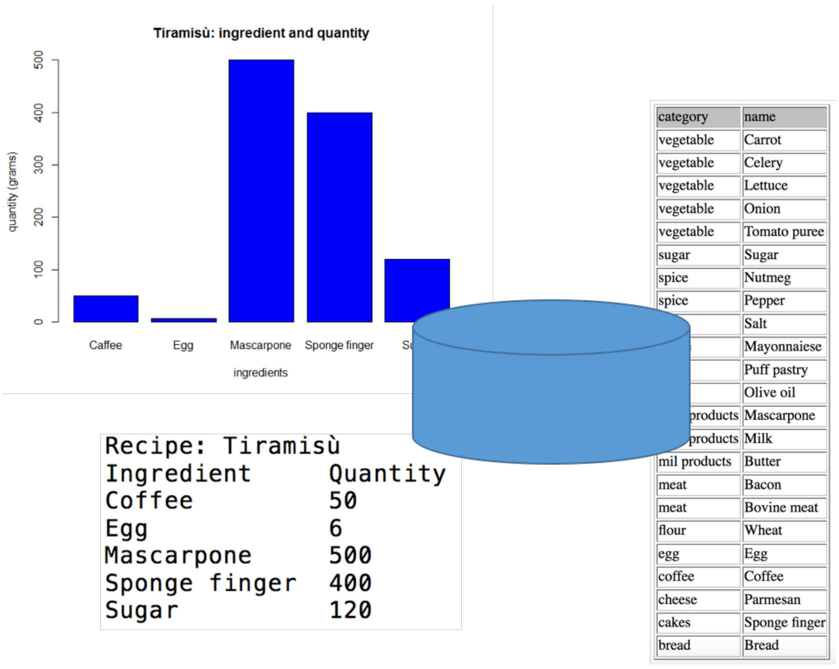
Database vs. File System

File System is the most convenient and fast way to store data on the hard disk, however files system provides only few features, specifically:

1. The **schema** for data is limited to the creation of a directory (folder in Windows jargon) structure for files;
2. Durability is not always supported nevertheless you back up data;
3. It is not supported a **query language** for data **into files**;
4. It does not support access to data from many users at once, in order to guarantee **consistency**.

Basic Concepts

Sharing Multiple Views of Data



```
Recipe: Tiramisù
Ingredient       Quantity
Coffee           50
Egg              6
Mascarpone       500
Sponge finger    400
Sugar            120
```

category	name
vegetable	Carrot
vegetable	Celery
vegetable	Lettuce
vegetable	Onion
vegetable	Tomato puree
sugar	Sugar
spice	Nutmeg
spice	Pepper
[illegible]	Salt
[illegible]	Mayonnaiese
[illegible]	Puff pastry
[illegible]	Olive oil
products	Mascarpone
products	Milk
mil products	Butter
meat	Bacon
meat	Bovine meat
flour	Wheat
egg	Egg
coffee	Coffee
cheese	Parmesan
cakes	Sponge finger
bread	Bread

Figure 1: Views from the database **Recipe**

Levels of Abstractions

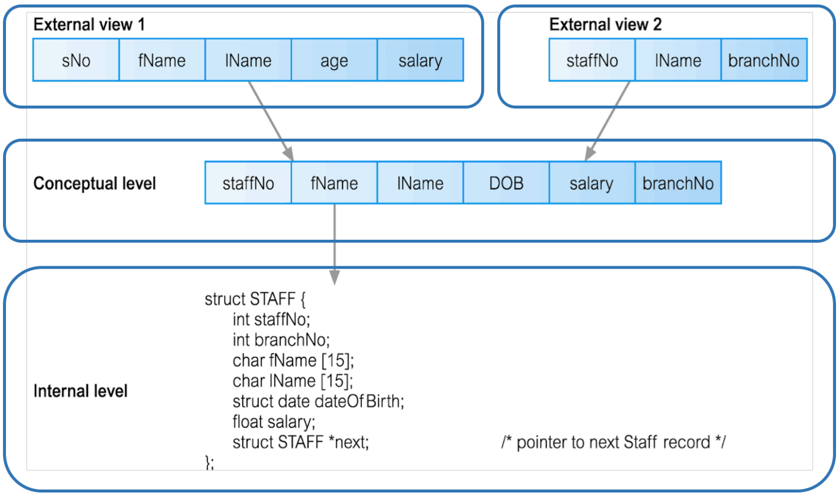


Figure 2: Working of the three level ANSI/SPARC [3]

ANSI/SPARC three-level database architecture model [3]: architecture layer which decouples external views on data and the implementation view of data.

(a) Many external views: describe different views of database for particular users or group of users;

(b) One (logical) schema: describes the logical data model;

(c) One physical (internal) schema: describes the storage structures of the database (how data is stored and accessed).

Database Management System

A **Database Management System (DBMS)** is a specialized software for creating and managing large amounts of data efficiently, and allowing it to persist over long periods of time, safely.

The Database Management System (DBMS) allows to:

1. Create new databases and specify their schemas (logical structure of the data), using the *data–definition language* (DDL);

2. Query the data and modify the data, using the *data–manipulation language* (DML), often called *query language*;

Queries and other actions can be grouped into *transactions*, which are units that must be executed atomically and in isolation

3. **Store** large amounts of data - even terabytes of data - over a long period of time;
4. **Recover** data in case of failures, errors or misuse, enabling durability;
5. **Control** access to data from many users at once.

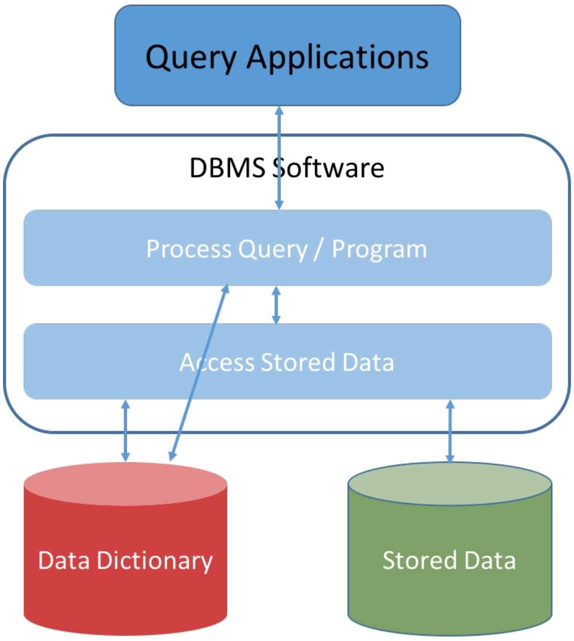


Figure 3: Database System

Our focus will be on `MySQL`, an open source database management system [5].

Trends

- The pervasiveness of the Web deals to the management of hierarchically structured data (XML).
- The continuous growing of data needs an Object-Oriented schema (NoSQL).

Applications

The most important applications of DBMS are, for example, in banking, airline systems, corporate record keeping, and in bigger systems that hold huge amount of data: indexing documents (google), pictures repository (Instagram), movie (YouTube), ... and moreover.

Today DBMS relevance is not strictly related to everyday business data, but mostly as a source of information and knowledge.

Data, Information and Knowledge [4]

1. **Data** generally represents a structured codification of single primary entities, as well as of transactions involving two or more primary entities
2. **Information** represents the outcome of extraction and processing activities carried out on data
3. Information becomes **Knowledge** when it is used to make decisions and develop the corresponding actions

Big Data

Despite the relevance of DBMS in many type of business, today they appear inadequate in the "**Big Data**" era.

Big data is essentially a massive amount of data that can be analyzed and used to make decision

Data is taken from a large variety of sources and different type of formats including **structured**, **semi-structured** and **unstructured** or combination of them.

- DBMS can only support structured data, but offer little support for semi-structured or unstructured data;
- DBMS scale up with expensive hardware, then unsuitable for growing data volume.

Big Data has value when used for deriving a decision using **analytics** techniques and methods:

1. **Text analytics**: the process of analyzing unstructured text, typically from social network, emails, blogs, etc., to extract from relevant information;
2. **Audio analytics**: the process of analyzing unstructured audio data, typically from call centers;
3. **Video analytics**: the process of analyzing unstructured video streams, typically from video surveillance;
4. **Social media analytics**: the process of analyzing structured and unstructured data from various social media sources: Facebook, LinkedIn, Twitter, YouTube, Instragram, Wikipedia, etc.
5. **Predictive analytics**: techniques for predicting future outcomes based on past and current data.

In business a special type of analytics is becoming more and more relevant **Web Analytics**:

- **Web analytics** is a way of learning how users interact with websites and mobile apps.
- **Web analytics tools** record pages a user views, when he or she views it, and in what order, returning useful information.

1. Web analytics tools stitch together the story of how each user moves through a website.
2. They capture how a user got to a website, such as by doing a **search in a search engine** or following a link **from another website**.
3. **Web analytics tools are mainly used for online marketing**, to introduce company's brand to people and enticing them to become customers.

Data Model

What is a data model? A data model is a notation for describing data and related information. It provides: *structure of data* (conceptual model - view), *operations on data* (high level), *constraints on the data* (limitations on data).

We aim to approach the two relevant data models:

- Relational Model - based on tables
- **Semistructured data model** - based on XML tree or graph

Design of Databases

Design a database means to answer to few questions:

1) *What kinds of **information** go into the database?*
2) *How should the **information** be structured?*
3) *What assumptions are made about types or values of **data** items?*
4) *How do **data** items connects?*

The process of designing a database then begins with the analysis of what information the database must hold and what are the relationships among components of that information

In order to visualize the *database schema* (conceptual model) a language or notation is needed.

- The traditional and popular approach is the *entity–relationship (E/R) model* **notation**.

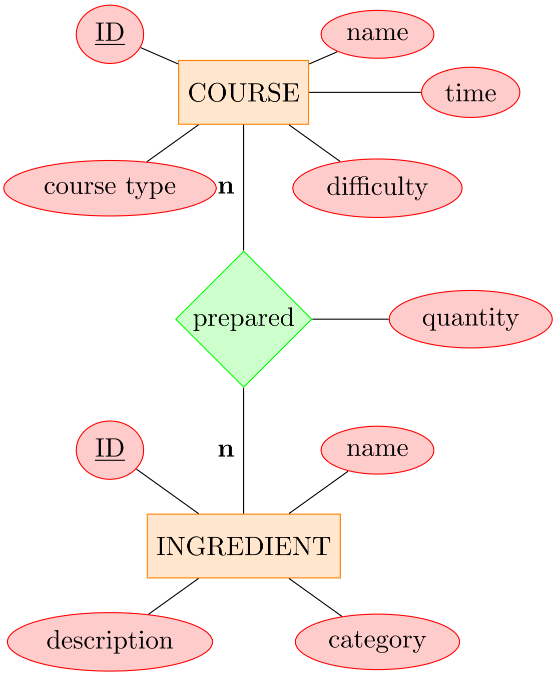


Figure 4: Recipe database schema

- The E/R model renders **data requirements** into a structure of data.
- The E/R model is a notation for describing only the **schema** of the database, not the database contents (= ***instance***), and not the operations on data.

Data Model: the Entity–Relationship Diagram

Elements of the E/R Model

In the **E/R model**, the structure of data is represented graphically, as an entity–relationship diagram, using three main element type:

1. **entity set**
2. **attribute**
3. **relationship**

Entity Set

An **entity** is an abstract object of some sort (student, employee, city, nation, ... , ingredient), and a collection of similar entities forms and **entity set** (students at the university, employees working for a company, cities of a nation, countries of a continent, ..., ingredients for cocking). *Commonly the name of an entity set is singular (student, employee, city, nation, ..., ingredient).*

Entity set is depicted by a rectangle.

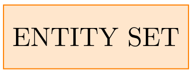


Figure 5: entity set, graphical shape

Attribute

Entity sets have associated **attributes**, which are common properties (or characteristics) of the entities in the set (name, surname, address, birth date, student ID, ... , name, capital, surface, flag, national song, ... , ingredient-name).

Attributes are depicted by ovals.

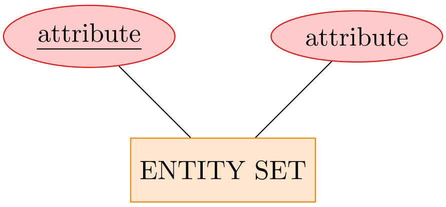


Figure 6: attribute, graphical shape

Keys are attributes or sets of attributes that **uniquely identify** an entity within its entity set (ID no. identifies a student, name identifies a nation, ingredient name identifies an ingredient, ...). Graphically the name of the attribute is underlined.

Relationship

Relationship is a connection among two entities of entity sets (student [*enrolled*] university, ... citizen [*live in*] city, ... course [*prepared with*] ingredient). Any number of entity sets could be involved in a relationship. *Commonly the name of the relationship comes from a verb (enrolled (to enroll), live in (to live), work for (to work), belonging (to belong), prepared with (to prepare)...).*

Relationship is depicted by a diamond, connected to entity sets.

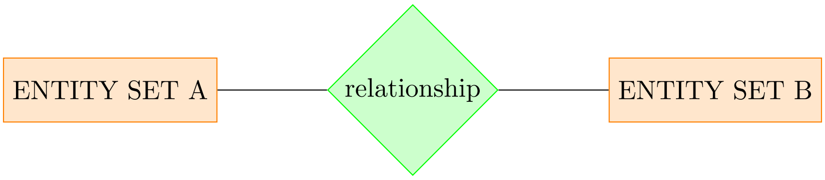


Figure 7: relationship, graphical shape

Sometimes it is convenient, or even essential, to associate attributes to a relationship, rather than with any one of the entity sets that the relationship connects (*belonging* - from, *enrolled* - year, *booked* - quantity, *recipe* - quantity...).

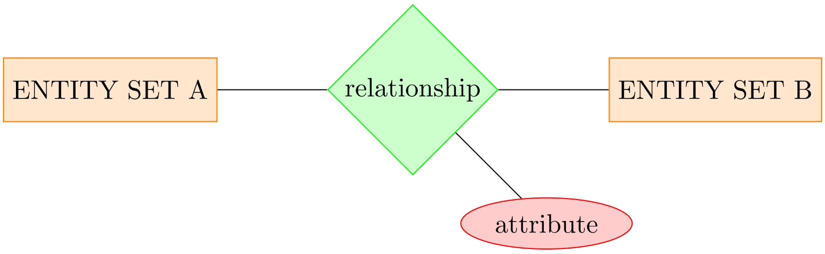


Figure 8: relationship, graphical shape

Given an *instance* of the database (data it holds), a **relationship** which connects entity sets, has an **instance** that consists of a finite set of **entities** chosen from the instance of each entity set that partecipate to the relationship. We regard to this relationship as the **relationship set** for the **database instance**.

For example: **{(John, unibo), (Mark, unibo), (Alice, unive), (Anne, polimi)}**

Entity–Relationships Diagram

An E/R diagram is a graph made up by entity sets, attributes, and relationships connected by lines

Multiplicity of Binary E/R Relationship

Assume **R** a relationship connecting the entity sets **A** and **B**:

- If each entity of **A** can be connected by the relationship **R** to at most one entity of **B**, then **R** is **many-one** from **A** to **B** (*many* students enrolled at *one* university, *many* citizens live in *one* nation). Similarly if each entity of **B** can be connected by the relationship **R** to at most one entity of **A**, then **R** is **one-many** from **A** to **B** (in *one* city live *many* students).

Multiplicity is indicated beside the entity set boxes with different types of notation and information. For simplicity here **n** (or **m**) indicates the many side of **R**, while **1** indicates the one side of **R**

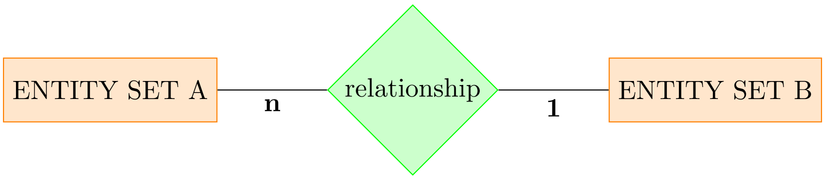


Figure 9: many-one

- If an entity of either entity set can be connected to at most one entity of the other entity set, then **R** is **one-one** (*one* city is the capital of *one* nation);

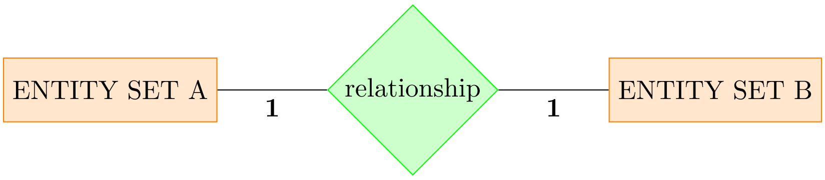


Figure 10: one-one

- If an entity of either entity sets can be connected to **many** entities of the other entity set, then **R** is **many-many** (*many* students attend *many* courses);

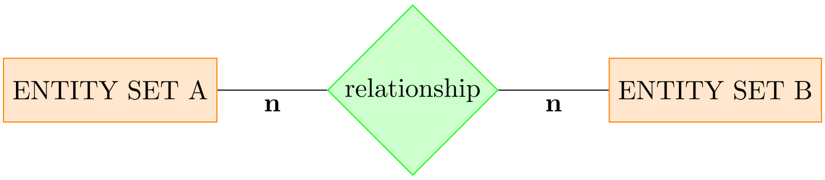


Figure 11: many-many

More Relationships

Multiway Relationship

Sometimes it's convenient to define relationships involving more than two entity sets. A multiway relationship in an E/R diagram is represented by lines from the relationship diamond to each of the involved entity sets. (*many* students [*attend*] *many* courses at *one* university).

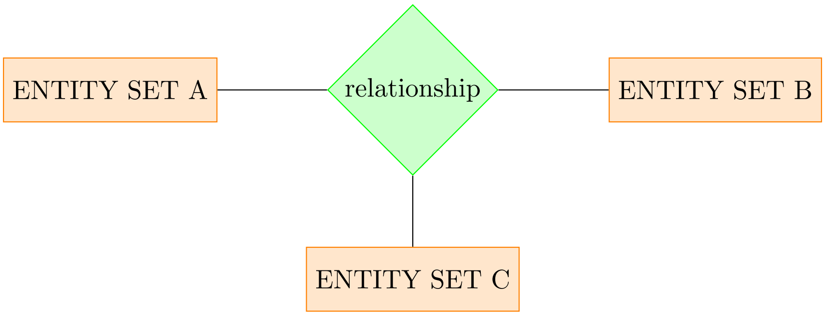


Figure 12: multiway relationship

Roles in Relationship

It is possibile that one entity set can appears two or more times in a single relationship. Many lines from the relationship to the entity set appear in the relationship. Each line represents a different *role* that the entity set plays in the relationship. A label indicates the *role* (*many* citizen [*governed by*] *one* citizen: roles *citizen*, *mayor*).

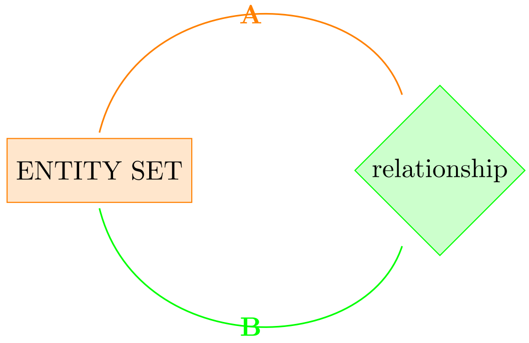


Figure 13: roles A and B in relationship

Parallel Relationships

E/R diagram can describes several relationship that connect the same entity sets, with distinct names, reflecting the different meanings of the relationships (citizen [*be a citizen of*] nation, citizen [*be elector*] nation).

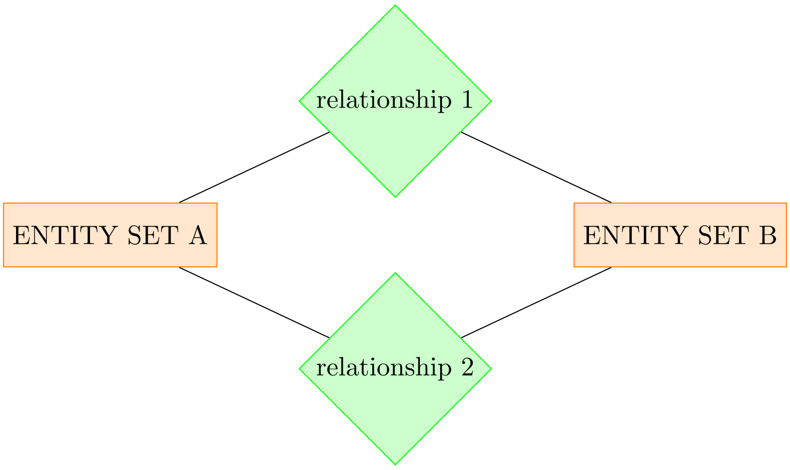


Figure 14: parallel relationships

E/R Database Design

For the following database proposals, design the corresponding E/R diagram. Draw entity sets with their attributes, relationships with their multiplicity, state key attributes.

Ex. # 1

"High-quality Italian food marketplaces, are now available on the Web. High quality regional Italian food can be purchased with few clicks and then tasted. This is a way to exhibit the culture and history of Italian regions".

Assume to design a database in order to keep track of traditional foods to sell by e-commerce web site.

Draw the E/R diagram that capture the requirements stated below:

1. Italian traditional **food** is described through a name, weight and unit, which uniquely identify a food product.

   [Examples: **Red Orange Juice, 500, ml**; **Red Orange Juice, 1, L**].

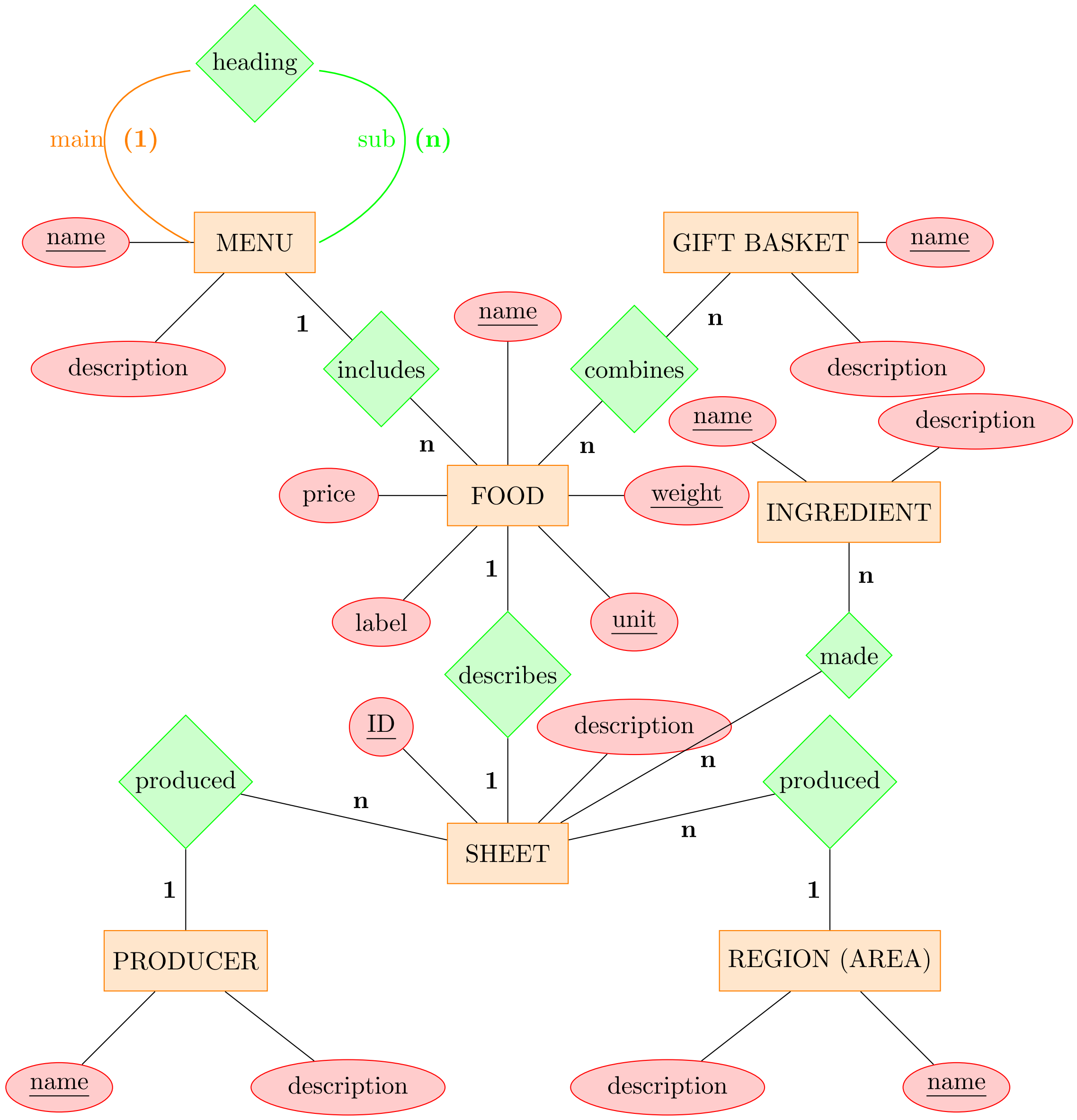
   Further it has a price and a descriptive label.

2. In order to give a user friendly tool for searching, food products are grouped. We assume to define a **menu** of items, each one having a name and a complete description, where each item represents a group of foods. Aiming to improve searching, *menu items* can be grouped into *main menu item*, for example "salumi" and "formaggi" can be items of the main menu "salumi e formaggi". This means to define a relationship between menu items.

3. Combinations of different foods are prepared as **gift baskets**, especially in some periods of the year (for example Christmas). The database keeps track of all gift baskets, simply with a name and a description.

4. Food is described by a **sheet**, that can be shown partially or fully. In this design it is reasonable to register and show a unique sheet for each product. A sheet has at least one description, even thought it can be related to many other data providing additional information.

5. **Producers (brand)**, **ingredients** and **regions (areas)** are registered and can be consulted as additional information of a food sheet. Basically they have a name and a description. **Further we assume food local area specificity.**


heading
main
(1)
sub
(n)
name
MENU
GIFT BASKET
name
1
n
description
includes
name
combines
description
name
description
n
n
price
FOOD
weight
INGREDIENT
1
n
label
unit
describes
made
ID
description
n
produced
n
1
produced
SHEET
n
1
1
PRODUCER
REGION (AREA)
name
description
description
name

Ex. # 2

"Mobility by bicycle represents one of the most relevant challenge to preserve the environment and improve life quality. Currently most of the cities are moving to the new paradigm of **Smart Cities**, realizing or improving **Bicycle Paths**". Most of the information about paths, could be consulted by an 'app' for smartphone.

The goal is to design a database to support this 'app', managing a variety of information on urban bicycle paths.

Draw the E/R diagram that capture the requirements stated below.

1. Many **bicycle paths** allow individuals to reach different points of the city. Each one has a code (initials), a color, and some characteristics.

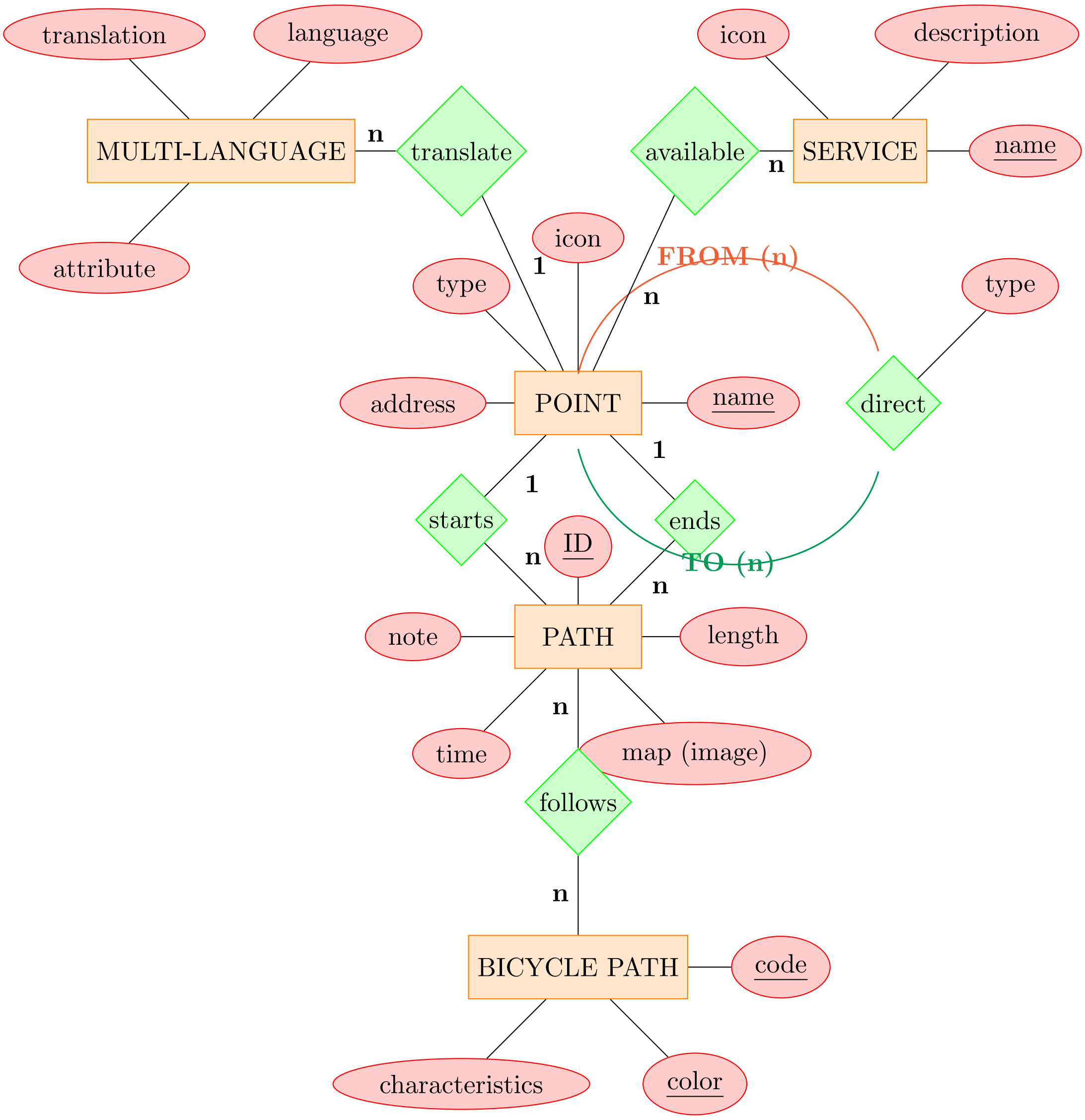
2. A single bicycle path follows many *basic neighboring small* **paths**, which typically join two relevant city points (university, theatre, swimming pool, bridge, square, etc.). To cycle along a path it is helpful to have the map (image), to know the total length, and time needed on average, and additional notes describing the path.

3. City **points** mainly represent locations which could be *the departure* or *arrival* for a path, but not only. It has a name, an address, a type, because it could be a public building, like university, theatre, post office, railway station, etc. or simply a square, bridge, roundabout. Generally an icon is used for a visual representation of the point (if applicable).

4. From a *point* it could be possibile to reach other *points*, not only through the path but for example walking, or cycling on ordinary streets. This type of information is very helpful and therefore it should be captured in the database as a *special relationship.*

5. Frequently at points which are buildings are associated few **services**, for example at the sporting centre could be practiced swimming, athletics, etc., moreover at the shopping centre it is possible to go to the cinema. A service has a name, a description, and could be visualized by an icon.

6. To be effective the 'app' should provide information in different languages, therefore **multi-language** translations of points information are managed. Specifically for the *attributes of point: name, address, and type* and for specified languages the corresponding translations must be provided.


translation
language
icon
description
MULTI-LANGUAGE
n
translate
available
n
SERVICE
name
attribute
icon
1
FROM (n)
type
n
type
address
POINT
name
direct
1
1
starts
ID
ends
n
TO (n)
n
note
PATH
length
n
time
map (image)
follows
n
BICYCLE PATH
code
characteristics
color

Ex. # 3

“enjoy@fair **application** advises tourist who likes enjoying in the spare time at the traditional fairs and festivals.” Thanks to the functionalities offered the user browses all events which could be regional fairs or festivals, even to consider selections by theme, season ... area. The goal is to design a database supporting the ‘app’.

Draw the E/R diagram that capture the requirements stated below.

1. All available information on fairs and festivals is collected and registered as **event**, specifically for each one type, title and in case a sub-title, mention about their history. Considering that an event is either a fair or a festival or both, and that the related information could vary we aim to distinguish it.

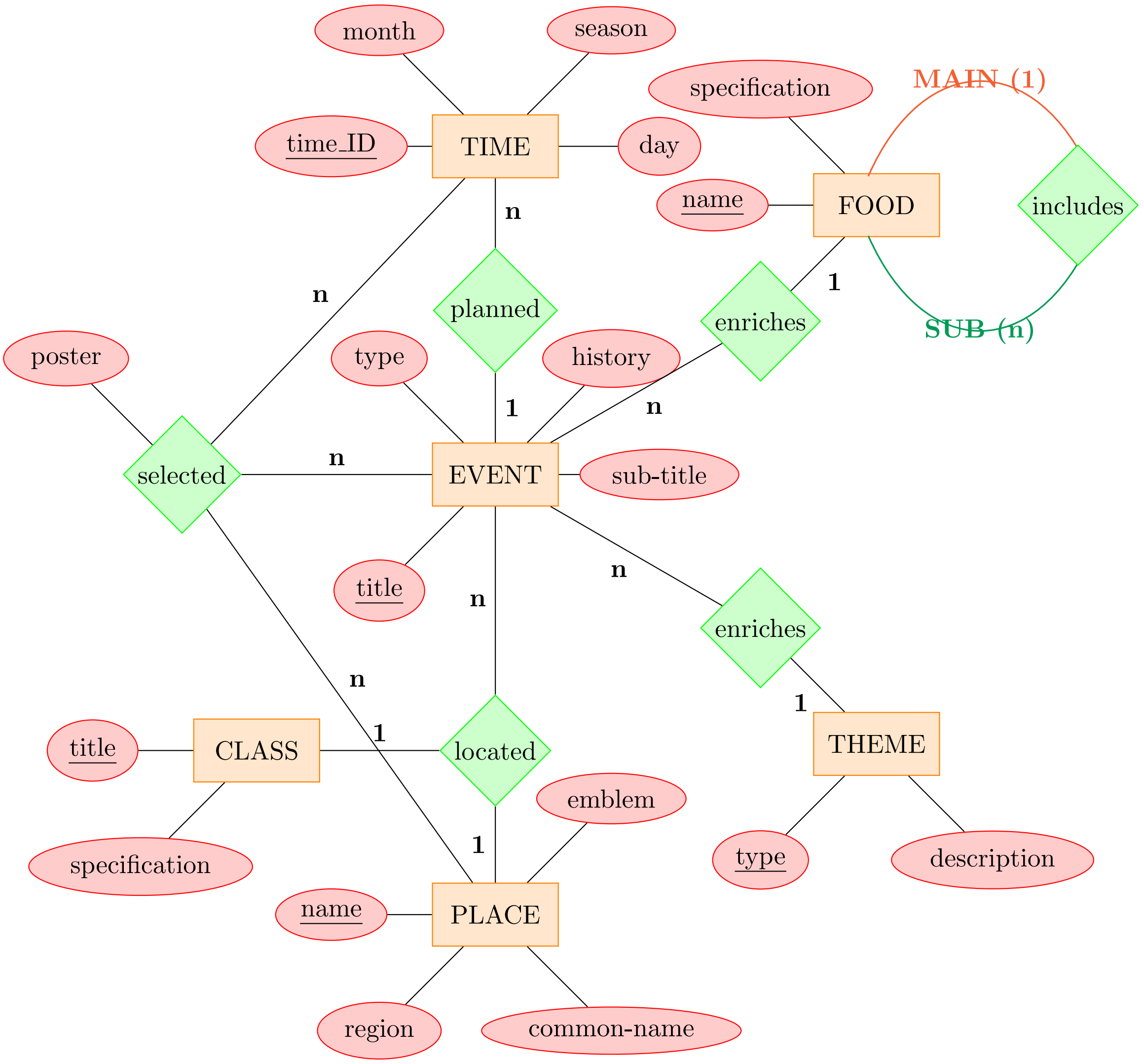
2. Firstly **food**, with its name and specification, which generally specifies fairs (for example jam fair, truffle fair, pork fair, ...). *In few cases we have that many foods would be registered with a generic name, for example pork, which includes, ham, salami, .. this fact has to be captured.* Then **theme**, with type, for example harvest, patron saint,.. and at least a description. These registrations (both) should integrate information of *event*.

3. Event has to be integrated with more information. **Time** registers information about when the event holds, specifically the season, month and day. An event could be held many times in one season or year. Identify a proper and informative key.

4. Place completes the set of information. **Place** holds name, common name, region, and emblem of the village/city.

5. Event and place information is sufficient to assign a **class** to the event (for example Bologna *antiques* fair, Reggio *farm* festival, ...). Class holds title and specification.

6. Selections of events are generally promoted, creating a *poster* showing the list of events, with place and time they holds. **The poster, which is a document, is registered**.


month
season
specification
MAIN (1)
time_ID
TIME
day
name
FOOD
includes
n
n
planned
1
enriches
SUB (n)
poster
type
history
1
n
selected
n
EVENT
sub-title
title
n
n
enriches
n
1
title
CLASS
1
located
THEME
emblem
specification
1
type
description
name
PLACE
region
common-name

References

1. J. Ullman, J. Widom, A First Course in Database Systems - Third Edition, Pearson - Prentice Hall
2. P. Atezeni et Altri, Basi di dati: modelli e linguaggi di interrogazione, McGraw Hill
3. D. C. Tsichritizis, A. Klug (Eds), The ANSI/X3/SPARC DBMS framework: report of the study group on database management systems, Information Systems 3 (1978)
4. C. Vercellis, Business Intelligence - Wiley
5. A. Meier and M. Kaufmann, SQL & NoSQL Databases - Springer
6. `http://www.mysql.com`, an open source database