CSIT115 L2

Database Design Process

A simplified process of database design consists of the following stages:

- Conceptual modeling
- Conceptual modeling transforms a specification of a database domain into a conceptual schema
- Logical design
- Logical design transforms a conceptual schema into a logical schema, for example the headers of relational tables
- Physical design
- Physical design determines the implementation details and adds to a logical schema persistent storage structures that improve performance, for example the indexes, clusters, partitions, materialized views, and the others

Database Domain

A database domain is a selected fragment of the real world to be described by the contents of a database

For example, a typical simple business domain can be described as a sequence of statements:

- A company would like to store and to maintain information about its suppliers and the parts shipped by the suppliers
- A supplier is described by a supplier name, date of birth, salary, and city he/she lives in
- A part is described by a part number, part name, colour and price
- A shipment is described by a supplier number, part name and the quantity of ordered parts
- A supplier is identified by its supplier number and a part is identified by its part number

Database Schema

A database schema is a description of stored data expressed in the terms of a particular data abstraction level

- For example, a conceptual schema is a description of stored data expressed in the terms of classes objects, properties of objects, identifiers of classes of objects, associations between the classes of objects, multiplicities of attributes and associations and generalization hierarchies
- For example, a logical schema is a description of stored data expressed in the terms of attributes, values of attributes, rows, columns, headers and tables
- For example, a physical schema is a description of stored data expressed in the terms of files, indexes, clusters, partitions, materializations, segments, extents and data blocks

Object Modeling

Object modeling is a special kind of conceptual modeling where a specification of a database domain is transformed into a simplified class diagram (conceptual schema)

Principles of object modeling:

- Contents of a database is quantised into discrete objects
- Objects are described by the attributes (properties) and operations (methods)
- Good news: we ignore the operations !
- Objects are identified by the values of selected attributes
- A class of objects is a group of homogeneous objects with common properties, common semantics and common identifiers

Examples:

- A student is an object, a lecturer is an object, a lecture hall is an object, a shipment is an object, an accident is an object, etc
- A student is described by the attributes student number, first name, last name, date of birth, etc
- A student is identified by a student number, a lecture hall is identified by building number and room number, a shipment is identified by a supplier name, date, and time, etc
- A group of students forms a class STUDENT, a group of lecturers forms a class LECTURER, a group of shipments forms a class SHIPMENT, etc

More object modeling concepts:

- A link a conceptual connection between two or more objects
- An association represents a group of homogeneous links with a common structure, common attributes, common semantics and common identifiers

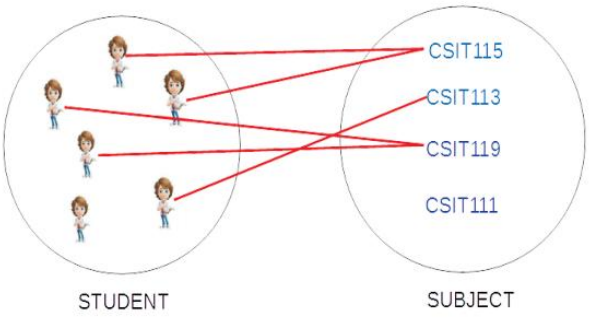
Example of links:

- James talks to Janusz
- Lecture 1 in CSIT115 is in building 3 room 2
- Peter supplies bolts to James

Examples of associations:

- STUDENT Talks-to LECTURER
- LECTURE Is-in BUILDING
- SUPPLIER Supplies PART To MANUFACTURER

Visualizations of objects, links, classes, and associations



More object modeling principles:

- A generalization hierarchy represents Is-a-subset relation between the classes of objects
- If a set of all objects in a class X is a subset of a set of all objects in a class Y then class Y is a generalization of class X
- In other words, if a class Y is a generalization of class X then a set of all objects in Y includes a set of all objects in X

Example of generalization:

- A class STUDENT is a generalization of classes UNDERGRADUATE STUDENT and POSTGRADUATE STUDENT
- This is so, because a set of all undergraduate students is a subset of a set of all students and a set of all postgraduate students is a subset of a set of all students
- In other words, a set of all students includes a set of all postgraduate students and it also includes a set of all undergraduate students

Another example of generalization:

- A class HUMAN is a generalization of classes STUDENT and LECTURER
- This is so because a set of all students is a subset of a set of all humans and a set of all lecturers is a subset of a set of all humans
- In other words, a set of all humans includes a set of all students and it also includes a set of all lecturers

Yet another (... and it is my favorite ...) example of generalization:

- A class BAT is a generalization of classes GREY-BAT, VAMPIRE-BAT, and BATMAN
- This is so because a set of all grey bats is a subset of a set of all bats and a set of all vampire bats is a subset of a set of all bats and a set of all batmen is a subset of a set of all bats