ST. XAVIER’S COLLEGE

**(Affiliated to Tribhuvan University)**

Maitighar, Kathmandu



**Database Management System Assignment #4**

**Submitted by:**

Bishal Pandey  
013BSCCSIT016

**Submitted to:**

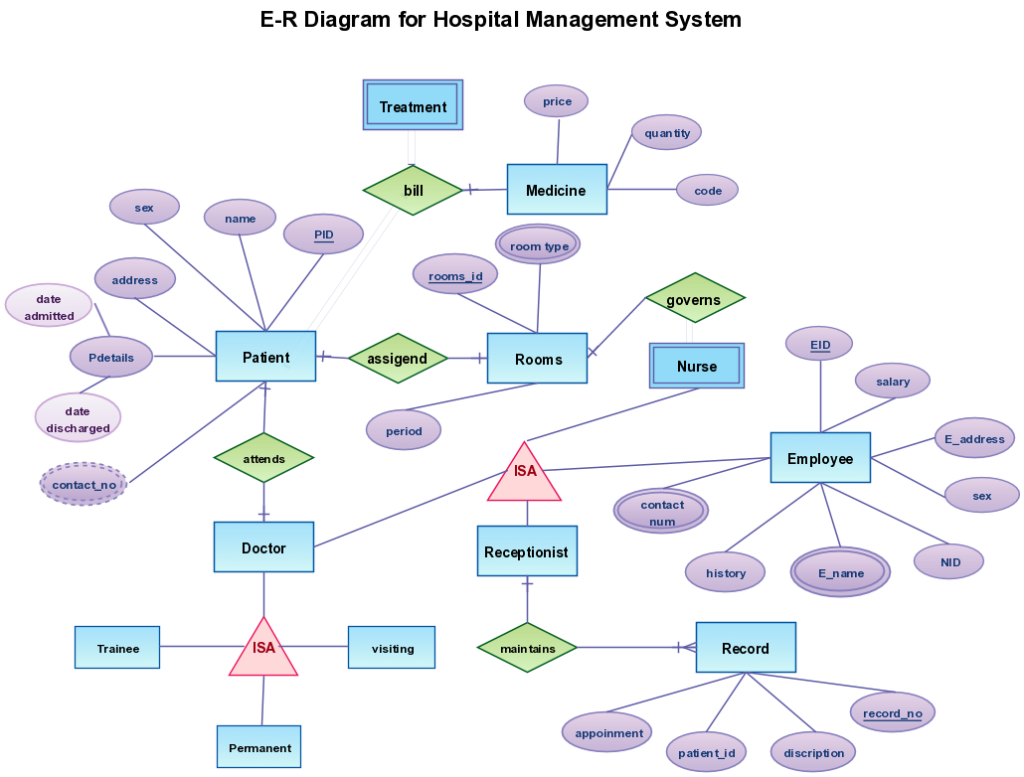
|  |  |
| --- | --- |
| Er. Sanjay Kumar Yadav Lecturer, St. Xavier’s College |  |

Theory Assignment#4

4.1 E.R-Diagram with one case study:

**Case study in Hospital Management system**

A General Hospital consists of a number of specialized wards (such as Radiology, Oncology, etc) .Information about ward includes unique name, total numbers of current patients. Each ward hosts a number of patients, who were admitted by a consultant (doctors) employed by the Hospital. On admission, the date and time are kept. The personal details of every patient includes name, Medical Recode Number (MRN), set of phone and one address (city, street, code). A separate register is to be held to store the information of the tests undertaken. Each test has unique episode No. , category and the final result of test. Number of tests may be conducted for each patient. Doctors are specialists in a specific ward and may be leading consultants for a number of patients. Each patient is assigned to one leading consultant but may be examined by other doctors, if required.



4.2 Design:

4.2.1 Functional Design:

4.2.2 Database Design:

**Database design** is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.

The three levels of data modeling, conceptual data model, [logical data model](http://www.1keydata.com/datawarehousing/logical-data-model.html), and [physical data model](http://www.1keydata.com/datawarehousing/physical-data-model.html), were discussed in prior sections. Here we compare these three types of data models. The table below compares the different features:

|  |  |  |  |
| --- | --- | --- | --- |
| Feature | Conceptual | Logical | Physical |
| Entity Names | ✓ | ✓ |  |
| Entity Relationships | ✓ | ✓ |  |
| Attributes |  | ✓ |  |
| Primary Keys |  | ✓ | ✓ |
| Foreign Keys |  | ✓ | ✓ |
| Table Names |  |  | ✓ |
| Column Names |  |  | ✓ |
| Column Data Types |  |  | ✓ |

Below we show the conceptual, logical, and physical versions of a single data model.

|  |  |  |
| --- | --- | --- |
| **Conceptual Model Design**  Conceptual Model Design | **Logical Model Design**  Logical Model Design | **Physical Model Design**  Physical Model Design |

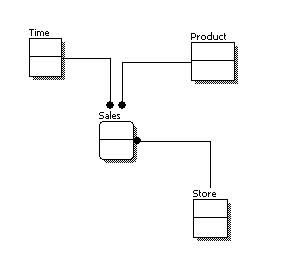
4.2.2.1 Conceptual DB design:

A conceptual data model identifies the highest-level relationships between the different entities. Features of conceptual data model include:

* Includes the important entities and the relationships among them.
* No attribute is specified.
* No primary key is specified.

The figure below is an example of a conceptual data model.

**Conceptual Data Model**



From the figure above, we can see that the only information shown via the conceptual data model is the entities that describe the data and the relationships between those entities. No other information is shown through the conceptual data model.

4.2.2.2 Logical DB Design:

A logical data model describes the data in as much detail as possible, without regard to how they will be physical implemented in the database. Features of a logical data model include:

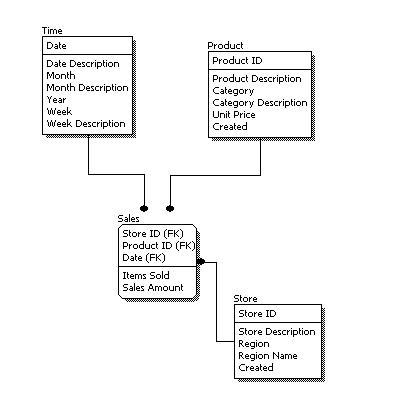
* Includes all entities and relationships among them.
* All attributes for each entity are specified.
* The primary key for each entity is specified.
* Foreign keys (keys identifying the relationship between different entities) are specified.
* Normalization occurs at this level.

The steps for designing the logical data model are as follows:

1. Specify primary keys for all entities.
2. Find the relationships between different entities.
3. Find all attributes for each entity.
4. Resolve many-to-many relationships.
5. Normalization.

The figure below is an example of a logical data model.

**Logical Data Model**



Comparing the logical data model shown above with the [conceptual data model](http://www.1keydata.com/datawarehousing/conceptual-data-model.html) diagram, we see the main differences between the two:

* In a logical data model, primary keys are present, whereas in a conceptual data model, no primary key is present.
* In a logical data model, all attributes are specified within an entity. No attributes are specified in a conceptual data model.
* Relationships between entities are specified using primary keys and foreign keys in a logical data model. In a conceptual data model, the relationships are simply stated, not specified, so we simply know that two entities are related, but we do not specify what attributes are used for this relationship.

4.2.2.3 Physical DB design:

Physical data model represents how the model will be built in the database. A physical database model shows all table structures, including column name, column data type, column constraints, primary key, foreign key, and relationships between tables. Features of a physical data model include:

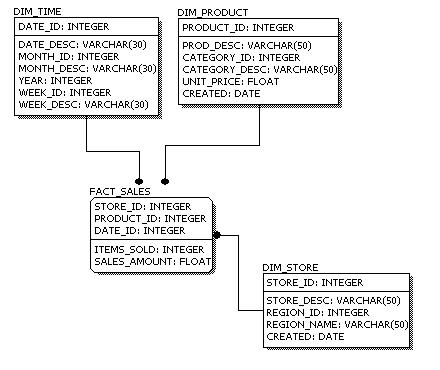
* Specification all tables and columns.
* Foreign keys are used to identify relationships between tables.
* Denormalization may occur based on user requirements.
* Physical considerations may cause the physical data model to be quite different from the logical data model.
* Physical data model will be different for different RDBMS. For example, data type for a column may be different between MySQL and SQL Server.

The steps for physical data model design are as follows:

1. Convert entities into tables.
2. Convert relationships into foreign keys.
3. Convert attributes into columns.
4. Modify the physical data model based on physical constraints / requirements.

The figure below is an example of a physical data model.

**Physical Data Model**



Comparing the physical data model shown above with the [logical data model](http://www.1keydata.com/datawarehousing/logical-data-model.html) diagram, we see the main differences between the two:

* Entity names are now table names.
* Attributes are now column names.
* Data type for each column is specified. Data types can be different depending on the actual database being used.

4.3 Characters of relation:

4.4 ER to relational mapping algorithm:

4.4.1 Mapping of regular entity Type:

4.4.2 Mapping of week entity Type:

4.4.3 Mapping of binary 1:1 relational type:

4.4.4 Mapping of binary 1:n relational type:

4.4.5 Mapping of binary m:n relational type:

4.4.6 Mapping of multivalve attributes:

4.6.7 Mapping of N-array relationship type: