ST. XAVIER’S COLLEGE

**(Affiliated to Tribhuvan University)**

**Maitighar, Kathmandu**

****

**Database Management System**

**Theory Lab Assignment #4**

**SUBMITTED BY:**

**Siddhant Rimal**

**013BSCCSIT039**

**SUBMITTED TO**

|  |  |
| --- | --- |
| **Er. Sanjay Kr. Yadav**  **( Lecturer )** |  |
| **Department of Computer Science** | |

Submission Date: August 12th 2015

1. **ER Diagram with one case study**

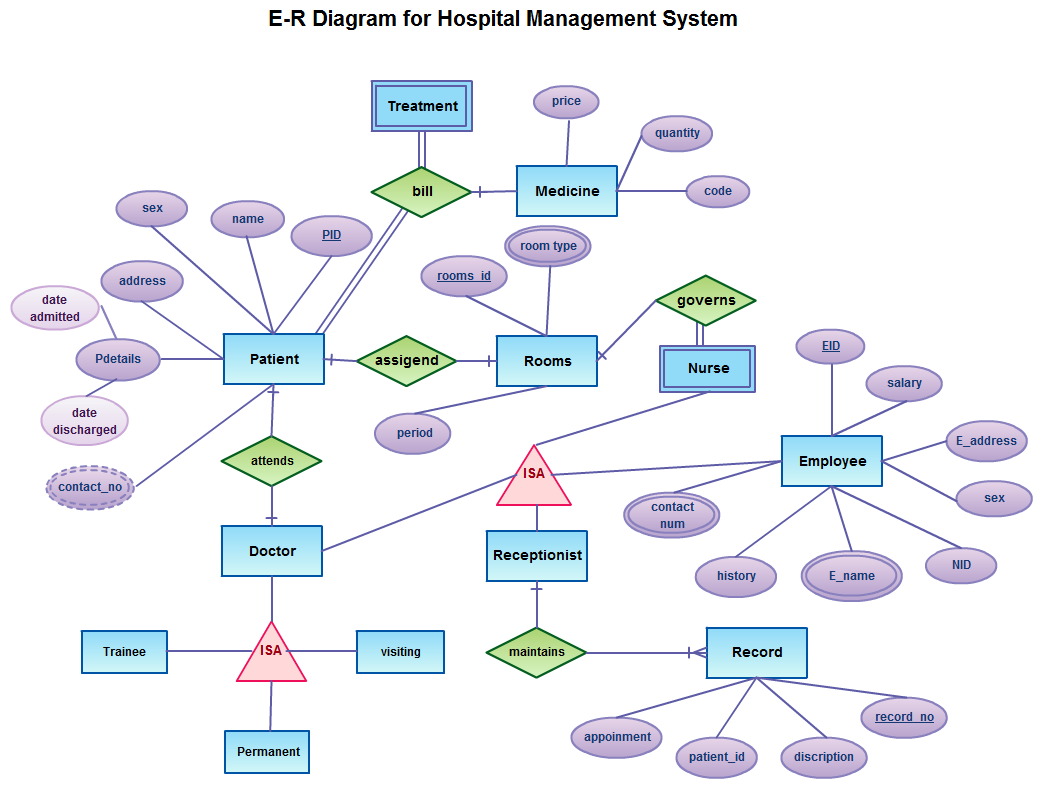
****

Fig.4.1: E-R Diagram of Hospital Management System

The system design of the hospital management system can be defined as follows:

* 1. **Patient:** This module handles all the data records maintained for the information about patient such as Patient ID, Name, and Address etc.
  2. **Rooms:** This module handles all the information about the availability and maintenance of hospital rooms. It deals with room type and assigned rooms to respective patients.
  3. **Doctor:** This module deals with the information records about doctors working in the hospital. It can form a relationship set with which patient the doctor should attend and such information.
  4. **Trainee:** This module deals with the information deals with the information about all the trainees that attend the hospital premises and use hospital resources for training purposes. They are handled by a doctor that is assigned to them.
  5. **Visiting:** This module deals with visiting and guest lectures provided to the trainees and staff.
  6. **Permanent:** This module deals with record about permanent staff.
  7. **Receptionist:** This module handles other relationship sets like ‘maintains’, which relates to Records module. It also keeps information about Nurse, Doctors and other Employee.
  8. **Record:** This module maintains hospital records about upcoming appointments, busy schedule, description of patient and problem etc.
  9. **Nurse:** This module has total participation in ‘governs’ relationship set. Thereby, it is responsible to govern the assignment of rooms.
  10. **Employee:** This module stores information about other employees of the hospital. Employee records like Employee ID, salary, Name, Designation is kept.
  11. **Treatment:** This module has total participation in ‘bill’ relationship set. Thereby treatment only occurs when patient pays the bill.

1. **Design**
   1. **Functional Design**

Functional Design is a paradigm used to simplify the design of hardware and software devices such as computer software and increasingly, 3D models. A functional design assures that each modular part of a device has only one responsibility and performs that responsibility with the minimum of side effects on other parts. Functionally designed modules tend to have low coupling.

The advantage for implementation is that if a software module has a single purpose, it will be simpler, and therefore easier and less expensive, to design and implement.

Systems with functionally designed parts are easier to modify because each part does only what it claims to do.

Since maintenance is more than 3/4 of a successful system's life, this feature is a crucial advantage. It also makes the system easier to understand and document, which simplifies training. The result is that the practical lifetime of a functional system is longer.

In a system of programs, a functional module will be easier to reuse because it is less likely to have side effects that appear in other parts of the system.

* 1. **Database Design**
     1. **Conceptual Database Design**

It is a process of constructing a data model for each view of the real world problem which is independent of physical considerations.

This step involves:

* Constructing the ER Model
* Check the model for redundancy
* Validating the model against user transactions to ensure all the scenarios are supported
  + 1. **Logical Database Design**

It is a process of constructing a model of information, which can then be mapped into storage objects supported by the Database Management System.

This step involves:

* Table Generation From ER Model
* Normalization of Tables
  + 1. **Physical Database Design**

The physical design of the database specifies the physical configuration of the database on the storage media. This step involves describing the base relations, file organizations, and indexes design used to achieve efficient access to the data, and any associated integrity constraints and security measures.

1. **Characteristics of Relations**

* No Duplicate Tuples - A relation cannot contain two or more tuples which have the same values for all the attributes. i.e., in any relation, every row is unique.
* Tuples are unordered - The order of rows in a relation is immaterial.
* Attributes are unordered - The order of columns in a relation is immaterial.
* Attribute Values are Atomic - Each tuple contains exactly one value for each attribute.  
  It may be noted that many of the properties of relations follow the fact that the body of a relation is a mathematical set.

1. **ER to relational mapping algorithm**
   1. **Mapping of regular entity types**

* For each entity type E in the ER schema, create a relation R that includes all the simple attributes of E.
* Add only simple components from any composite attributes in E.
* Choose one of the key attributes of E to be a primary key of R.



Fig4.4.1: Mapping of regular entities

* 1. **Mapping of weak entity types**
* For each weak entity type W with owner type E create a new relation RW that includes all the simple attributes of W as attributes of RW.
* In addition include a foreign key reference to the key of the translation RE of E.
* The key of RW will be the key of foreign key together with the mapped partial key from W.

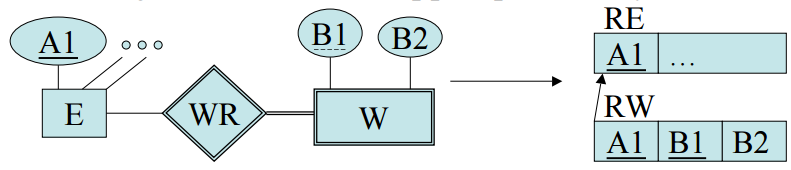


Fig4.4.2: Mapping of weak entities

* 1. **Mapping of binary (1:1) relationship type**
* For each binary 1:1 relationship R in the ER Schema, identify the relations S and T that correspond to the entity types participating in R.
* There are three possible approaches:
* The foreign key approach: Choose one of the relations, say S, and include in S a foreign key reference to the primary key of T. (Favor S over T if its corresponding entity participated totally in the relationship.)
* Merged relation approach: When both relations correspond to entities that participated totally in the relationship, one can just merge the two relations into one.
* Relationship relation approach: Set up a new relation with for the purpose of cross referencing the primary keys of tables S and T.
  1. **Mapping of binary (1:N) relationship type**
* For each 1:N binary relationship type R, identify the relations S and T corresponding to the entity types in this relationship.
* Further, suppose S is the N-side of the relationship.
* Then include a foreign key reference to R in S.

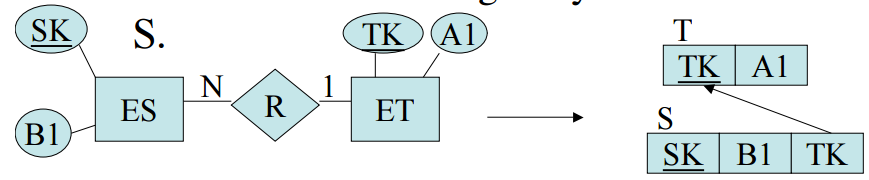


Fig4.4.4: Mapping of binary (1:N) relationship

* 1. **Mapping of binary (M:N) relationship type**
* For each binary M:N relationship type R, create a new relation RR in order to represent R.
* Include as foreign keys of keys the primary keys of the relations corresponding to the two participating entities.
* Also add as attributes simple attributes of R as in the entity case.

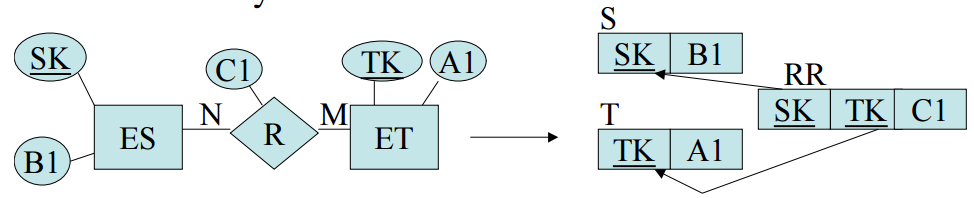


Fig4.4.5: Mapping of binary (M:N) relationship

* 1. **Mapping of multivalued attribute**
* For each multivalued attribute M, create a new relation RM. This relation will include an attribute corresponding to M as well as a foreign key reference to the relation corresponding to the entity that M was part of.
* The key will be both attributes.

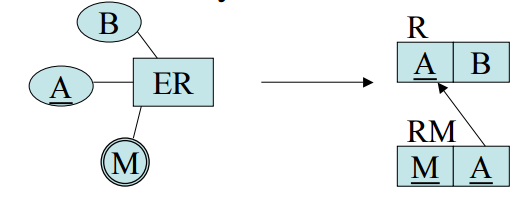


Fig4.4.6: Mapping of multivalued attribute

* 1. **Mapping of n-ary relationship types**
* For each n-ary relationship R with n >2, create a new relation RR in order to represent R.
* Include as foreign key attributes in RR the primary keys of the relations that correspond to the participating entities with cardinality constraints other than 1.
* The primary key of RR is the combination of these foreign keys.

1. **CONCLUSION**

* ER Diagrams play a very important role in the database designing process. They serve as a non-technical communication tool for technical and non-technical people.
* Entities represent real world things; they can be conceptual as a sales order or physical such as a customer.
* All entities must be given unique names.
* ER models also allow the database designers to identify and define the relations that exist among entities.

1. **REFERENCE**
2. “E-R Modelling”, Internet url: <http://www.guru99.com/er-modeling.html> 2015 [10/08/2015]
3. Safi Ullah, “Hospital Management System”, PPT, url: <http://www.slideshare.net/SafiUllah2/hospital-management-system-25384877> 2015 [10/08/2015]
4. KL University, Lecture notes, Internet url: <http://www.kluniversity.in/elearn/materials/tdbqucsbsb77152TDBQUCSBSB.pdf> 2015 [10/08/2015]
5. Elmasri, Ramez, & Navathe, Shamkant B., Fundamentals of Database Systems, Second Edition, Addison-Wesley Publishing Company, 1994, ISBN 0-8053-1748-1, pages 450-452.