ER Model - An Example

Chapter 2

in the Course Textbook

For material relating to ER Model

ramesh@cs.ubc.ca

http://www.cs.ubc.ca/~ramesh/cpsc304

Problem

A Bioinformatics Application

- *Patient*: has a unique *MSP number, a Patient name, a Date of Birth, a Tissue Type* and *an indicator* denoting whether the tissue is cancerous or normal.
- A *patient library* associates a patient with multiple *tags*
- Each tag has a unique *tag number* and a unique *nucleotide sequence*.
- For each tag in the patient library, a *count* is given to record the number of times the tag occurs in the library. In general, the same tag can be associated with any number of patients.
- A tag may be mapped to a *gene*. Each gene has a unique *gene name* and a *type*.
- In general, multiple tags may be mapped to the same gene. However, two different genes cannot be mapped to the same tag.
- Finally, an *article* is identified by a unique *article number* and a *journal name*. An article may analyze multiple genes and a gene may be analyzed by multiple articles.

Let us construct an ER model for the above application.

WARM UP

Entity Sets:

- Patients - Attibutes are MSP Number, Name, DOB, Tissue Type and Indicator
- Tags - Attributes are Tag Number and Nucleotide Sequence
- Genes - Attributes are Gene Name and Type
- Articles - Attributes are Article Number and Journal Name

Relationship Sets:

- Patient Library - Many to Many, Has an attribute Count
- Map - Many to 1 from Tags to Genes
- Analyzes - Many to Many

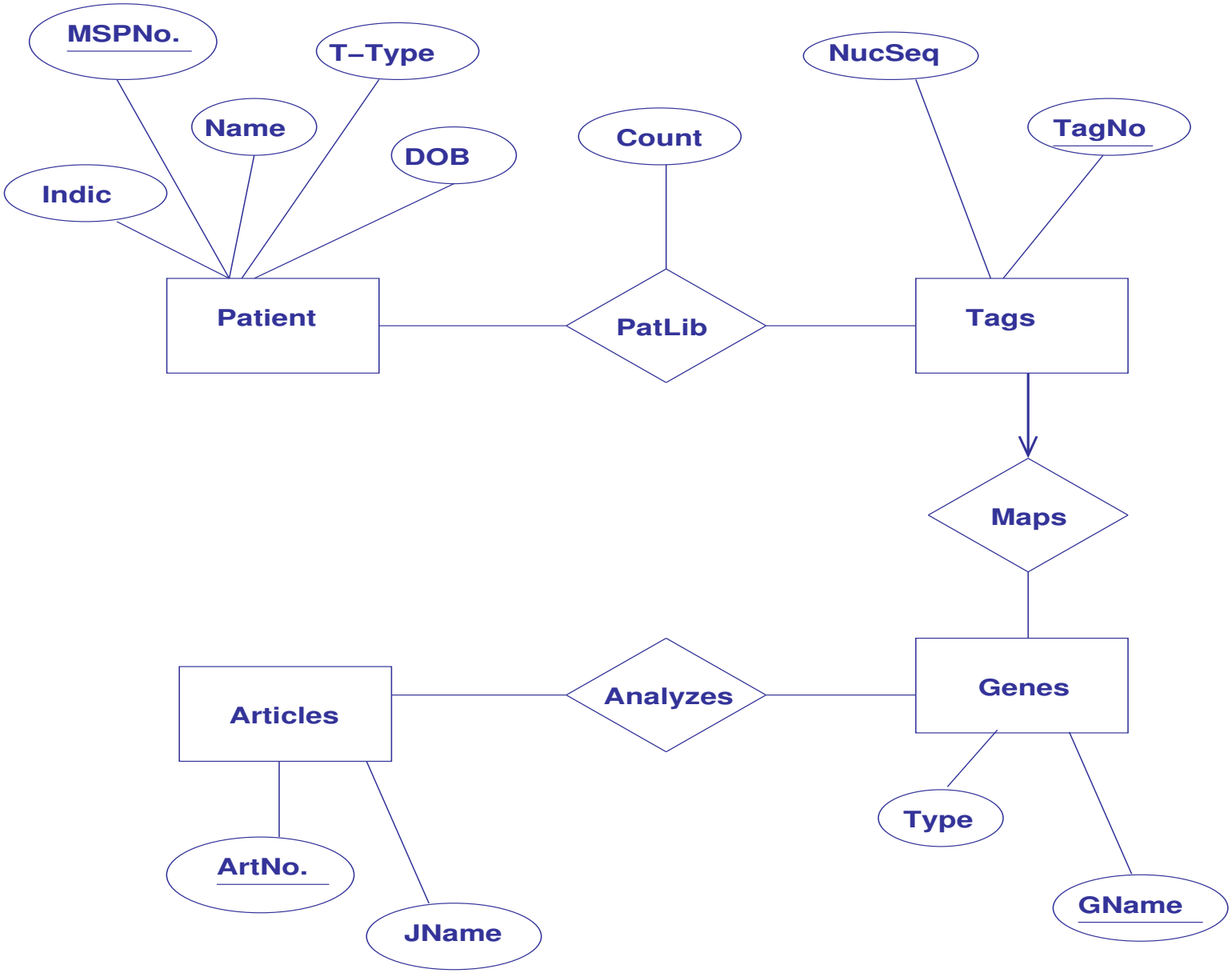

Figure 1: ER Diagram

Relations for the Entities

Patient Table: Patients(MSPNo, Name, TType, DOB, Indic)

```
CREATE TABLE Patient(
    MSPNo  CHAR(10),
    Name   CHAR(30),
    TType   CHAR(30),
    DOB     DATE,
    Indic    ENUM('Cancerous','Normal'),
    PRIMARY KEY (MSPNo));
```

Tags Table: Tags(TagNo, NucSeq)

```
CREATE TABLE Tags(
    TagNo  CHAR(10),
    NucSeq   CHAR(30),
    PRIMARY KEY (TagNo),
    UNIQUE(NucSeq));
```

Relations for the Entities

Genes Table: Genes(GName, Type)

```
CREATE TABLE Genes(
    GName   CHAR(30),
    Type     CHAR(30),
    PRIMARY KEY (GName));
```

Articles Table: Articles(ArtNo, JName)

```
CREATE TABLE Articles(
    ArtNo  CHAR(10),
    JName   CHAR(30),
    PRIMARY KEY (ArtNo));
```

Relations for the Relationships

Patients Library Table: PatLib(MSPNo, TagNo, Count)

```
CREATE TABLE PatLib(
    MSPNo    CHAR(10),
    TagNo     CHAR(10),
    Count     INTEGER,
    PRIMARY KEY (MSPNo, TagNo)
    FOREIGN KEY (MSPNo) REFERENCES Patient,
    FOREIGN KEY (TagNo) REFERENCES Tags);
```

Maps Table: Maps(TagNo, GName)

```
CREATE TABLE Maps(
    TagNo  CHAR(10),
    GName   CHAR(30),
    PRIMARY KEY (TagNo)
    FOREIGN KEY (TagNo) REFERENCES Tags,
    FOREIGN KEY (GName) REFERENCES Genes);
```

Relations for the Relationships

Analyzes Table: Analyzes(ArtNo, GName)

```
CREATE TABLE Analyzes(
    ArtNo    CHAR(10),
    GName    CHAR(30),
    PRIMARY KEY (ArtNo, GName)
    FOREIGN KEY (ArtNo) REFERENCES Articles,
    FOREIGN KEY (GName) REFERENCES Genes);
```

Tags and Maps can also be combined into one table as they share the same primary key, as follows.

```
CREATE TABLE TagsMap(
    TagNo    CHAR(10),
    NucSeq   CHAR(30),
    GName    CHAR(30),
    PRIMARY KEY (TagNo)
    FOREIGN KEY (GName) REFERENCES Genes,
    UNIQUE (NucSeq));
```