Homework 1

Due date: Wednesday, April 29, in class.

Problem 1

You are building a credit card transactions database. The following information is made available to you.

- The credit card company issues credit cards to individuals and corporations. Each credit card account is uniquely identified by its *number*. The database must also store information about the *date of issue* of the credit card, its *expiration date* (both are stored as *month, year* pairs), *billing address*, *phone number*, *credit limit*, *interest rate*, *date of last statement*, and *current ballance*.
- Each credit card account, both individual, and corporate allows for a number of authorized users. Each individual account is associated with a list of customers. Each corporate account is associated with a company and the company, in turn has a list of authorized users for the account, also customers.
- Individual customers may have multiple credit card accounts with the credit card company. Corporate customers have only one account.
- For each person (customer) the database records his/her *name* and *date of birth*.
- For each corporation, the database records its *name*, *corporate address* (may be different from the billing address on the account), and a contact phone number.
- The database records credit card transactions. Each transaction involves a specific account. For each transaction we store *transaction id* – the unique identifier of a transaction, *date it occurred*, *date is was posted*, *type* of transaction (credit, payment, cash advance, finance charge, fee), *vendor* and the *amount*.
- Information about vendors that accept credit cards is also stored in the database. Each vendor is uniquely identified by a *vendor id*. The database records the *name* of the vendor, the *billing address* of the vendor, the *type of business* of the vendor (e.g., retail, wholesale, services, etc) and the *transaction charge rate* (i.e., how much money, as a percentage of the transaction amount, the credit card company makes per transaction).

1. Specify all entity sets present in the database. For each entity set specify its attributes and indicate the primary keys.
2. Specify all relationship sets present in the database. For each relationship set specify the entity sets it associates with each other and any additional attributes that may be needed.
3. Specify any constraints present in the database.
4. Draw the ER diagram of the database you are proposing to build. (note: you do not need to indicate all attributes on the diagram, only primary keys, identifying attributes and relationship sets attributes will be enough)

Problem 2

Suppose, you are asked to make the following adjustments to the credit card database from the previous problem.

- Each credit card account can be of one of a number of types (e.g., "platinum", "gold", "student", "rewards", etc...). Each account type has a *name*, *allowed categories* (individual only, corporate only, both), *interest rate*, *fee amount* (the fee amount assessed for late payments), *annual fee* (can be 0), and a *description of any special conditions/promotions associated with the account type.*
- In addition, for accounts with corporate customers, the database now needs to store information about *the total number of transactions* to date. Similarly, for individual credit card accounts, the database stores the *number of customer service calls* and the *date of last call.*

1. Describe the changes in your database design. What new entity sets, relationship sets, if any, will be introduced? What new attributes? Are there any new constraints? What are the changes in the existing entity and relationship sets and constraints, if any?
2. Draw the E-R diagram for the modified database.

Problem 3

Consider the following small dataset descriptions. For each dataset:

(i) Identify all entity sets and their attributes.

(ii) Identify all relationship sets and their attributes.

(iii) Construct an E-R diagram. Specify any additional constraints that did not get reflected on the E-R diagram.

1. The dataset consists of a number of scientific articles published in various scientific journals. Each journal is identified by its *name.* For each journal, information about the number of issues per year is available. Each article is uniquely identified by its title and the journal it is published in. Additionally, we store information about the *year* of publication and the journal issue (a number) in which the article appeared. Articles can have multiple authors. Each author is uniquely identified by his/her full name (*first, last*) and *affiliation.*
2. The dataset is a collection of information from a social networking website. Website users have accounts, which are uniquely identified by the *account Id.* For each account, the dataset stores information about the user's real name (*first, last*), *email address*, *gender* and *birthdate.*

Users can be friends with each other. For each friendship, we store the date on which the friendship relationship was established. Friendship relationships are symmetrical. Users can also follow other users (we also store the date, this relationship was established). This relationship type is assymetrical. Users can make posts to their blog: the dataset records the *date* and *time* of the post and the full *text* of the post (assume the posts, a-la twitter, are limited in size to 512 characters). Posts can have tags associated with them: tags are assigned to the posts by users (authors, or others). Users can also write comments to the blog posts of others. For each comment, its *date*, *time* and *text* are stored. You can assume that both posts and comments come with unique ids (e.g., unique URL).

3. The dataset is a collection of information about the current U.S. Congress. Each Congressperson is either a U.S. Representative or a U.S. Senator. Senators represent individual states. Each state is represented by two Senators: senior and junior. Each Representative represent individual congressional districts within the states (e.g., CA-23). Senator/Representative names (*first*, *last*) are not unique, but their respective representation (state and seniority for Senators; state and district for Representatives) are. For each Congressperson, their *party affiliation* is also listed (typically, either Republican or Democratic, but there may be some other affiliation, i.e., Independent). Both the U.S. Senate and U.S. House of Representatives consider various bills, and members of appropriate houses vote on them. Each bill has a *name*, a unique *identification number* (e.g., `"H.Res.1"`), and may be sponsored by multiple Congresspeople from the chamber in which the bill is considered. We also record the date on which it was *introduced*. Each vote recorded in either chamber (Senate or House) is a consideration of a specific bill. For the purpose of this database, each bill is voted on at most once (i.e., there may be bills introduced w/o votes on them, but there cannot be a bill with more than one vote on it in chamber). For each vote, we record how each Congressperson voted on the bill. We also associate with each vote the final tally of the `Yes`, `No` and `Present` votes. Each Congressperson votes only in the votes in their chamber, and has at most one vote for each vote.

Problem 4

Note: This problem uses information from one of previous CSC 366 course projects. Generally speaking, you can find all information on how to design this database on the CSC 366 course wikis, However, I strongly recommend that you try solving this problem yourselves. (Note, that the data model is NOT the same as the one in the project, albeit it is similar).

Cal Poly's horse unit houses a few hundred animals at each moment of time. The animals are hosted in three different types of facilities: pastures, pens and stalls. A pasture is a large field, which houses multiple animals at the same time. A pen is a small, restricted area next to the horse unit proper, which can house multiple horses at the same time. A stall is a small restricted area that hosts at most one animal at a time. Stalls are combined into barns. While pens and pastures are simply referred to by their names, a stall is identified by the name of the stable it is in, and a stall number.

For each animal, the horse unit keeps track of its name, date of birth, breed (quarter horse, thoroughbred, etc), owner and a neck tag ID (a combination of 10 letters and digits). Name, breed and date of birth uniquely identify a horse, as does the neck tag ID. The horse unit keeps track of the locations in which each animal was housed. For each such location, the date the animal was moved to it and the date animal was moved out of it are collected and kept.

Horses eat a diet that consists of a number of different ingredients, including different kinds of hay as well as various minerals and vitamins. Each diet has a name (e.g., "Normal diet", or "Pregnant mare diet"), date of creation and the name of the person who created it associated with it. Each diet essentially describes the daily ration of an animal[1] The horse unit keeps tack of various ingredients (idenitfied by name, units of measurement and currently available quantity) available for inclusion in the diets. The diet is essentially the list of ingredients together with the quantity of each ingredient to be given to the animal.

Create an E-R model for the horse unit database described above.

Problem 5

Note: This problem uses information from one of previous CSC 366 course projects. Generally speaking, you can find all information on how to design this database on the CSC 366 course wikis, However, I strongly recommend that you try solving this problem yourselves. (Note, that the data model is NOT the same as the one in the project, albeit it is similar).

An e-commerce company desires to make its catalog of products available to on-line shoppers in multiple countries. Each product in the company's inventory is identified by its SKU (Stock-Keeping Unit) number, manufacturer, current availability information (available/not available), and available quantity. In addition, for each product, there are various text descriptions of it: product name, short product description, long product description and site review.

The textual data describing the product now needs to be present in the database in the language of each country in which the company wants to do business. Each country is identified by its name. Additionally, the continent and the URL of the company's e-commerce portal in that country need to be stored. As mentioned above, different languages are spoken in different countries. For each language, its name in English, its language group (e.g., "Germanic", "Romance", etc) and the name of the company employee responsible for translations into this language need to be kept track of. A language can be spoken in multiple countries (e.g., English is spoken in the US, UK, Australia and Canada), and a country can use multiple languages (e.g., Canada uses English and French, Ukraine uses Ukrainian and Russian, etc.) Product descriptions for two different countries may differ, even if the spoken language is the same - e.g., the text describing a product for the US and the UK audiences may be different.

Products must be properly priced for each country in which they are made available. Therefore, product prices must be availble in each currency used in each country. Note that products are priced independently in each country and the prices to do not depend on the currency conversion rates - therefore, specific prices for each currency must be present. For each currency, its name and a short abbreviation are stored. Each country has exactly one currency in which products are priced (e.g., US Dollars in the USA, British Pounds Sterling in the UK or roubles in Russia). Multiple countries can have the same currency (e.g., France, Germany, Italy all have Euro as the currency). Prices in the same currency for different countries may differ (e.g., the same item can cost 10 Euros in France and 10.75 in Germany).

Design an E-R model that accommodates the localization requiremnts on the products presented by the company.

Problem 6 *Exercise 4.1.9, textbook (both "A First Course" and "The Complete Book", 3d Ed.), page 140.*

Problem 7 *Exercises 4.2.5 and 4.2.6, textbook (both "A First Course" and "The Complete Book", 3d Ed.), pages 147-148.*

Problem 8 *Exercise 4.4.4 a), b) textbook, (both "A First Course" and "The Complete Book", 3d Ed.), page 157.*

[1]This was not true in the actual project. But for this assignment, we will assume it.