Dr. Stefan Brass
School of Information Sciences
University of Pittsburgh

July 10, 2001

INFSCI 2711 "Database Analysis and Design" — Example II for Midterm Exam —

Instructions

- This is the final exam for the Tuesday session of INFSCI 2711 in 1999. There was 1 hour and 30 minutes time to work on the exercises.
- Please note that in this course, there will be no midterm exam. The final exam will be 3 hours long and contain a part that is similar to this example plus a part about physical design.
- Note also that in 1999, I did not teach reverse engineering and UML class diagrams. These topics may appear in the real exam. The third example (to be distributed later) will contain such exercises.
- In general, there could be any new type of exercise. Distributing these exams does not mean that I guarantee that there are no other exercises. Please listen also to announcements in the last lecture before the exam.
- The exam will be "open book".
- In the multiple-choice questions, exactly one answer is correct.

Exercise 1 (ER Model) 8 Points

Consider a hospital or a healthcare center with several physicians (doctors). Our task is to develop an entity-relationship-schema for keeping track of prescriptions. The main question to be answered is “Who (i.e. which physician) prescribed what drug to which patient on which date?” In addition, we also want to store a short information about every drug containing e.g. the side effects and other warnings which might be important for the doctor to consider before he/she prescribes the drug.

We want to store historical information about prescriptions. It is possible that the same doctor prescribes the same drug to the same patient on different dates (renewal of prescription), and we need to store all these dates. Of course, it is also possible that the same doctor prescribes different drugs to the same patient on the same day. It is possible (although rare) that different doctors prescribe different drugs to the same patient on the same day. However, it is impossible that different doctors prescribe the same drug to the same patient on the same day.

For simplicity, the following diagrams show only the key attribute “Pat_ID” of patient. Of course, more data will be needed for patients, but this omission shall not count as an error in this exercise. In the same way, very little information is stored about Physicians and drugs. But for this exercise we concentrate only on the information requirement expressed in the above query.

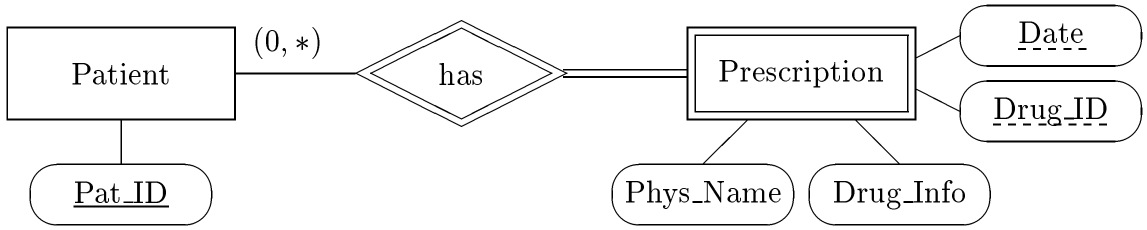
Which of the following ER-schemas are correct and which are not? If a schema is incorrect, please give an explanation.

- A schema counts as incorrect already if it is too general, i.e. allows database states which correspond to no legal state in the real world. In such a case, additional constraints would be needed.
- A schema counts as incorrect also if it contains redundant data.
- Of course, a schema is incorrect also if it cannot answer the question mentioned above.
- Of course, also ER-model syntax errors are possible.
- However, bad style alone does not qualify a solution as wrong. If you wish, you can comment on the style, but only exceptional remarks will earn extra credit.

Each subexercise is worth two points. If you are unsure about your answer, you can give an explanation what are the critical issues. You might then get at least one of the two points, but you might also lose a point if your comments are wrong.

Some exercises might contain multiple errors. In such cases it is sufficient to explain one error. In exceptional cases, very good explanations of all errors might earn a bit extra credit.

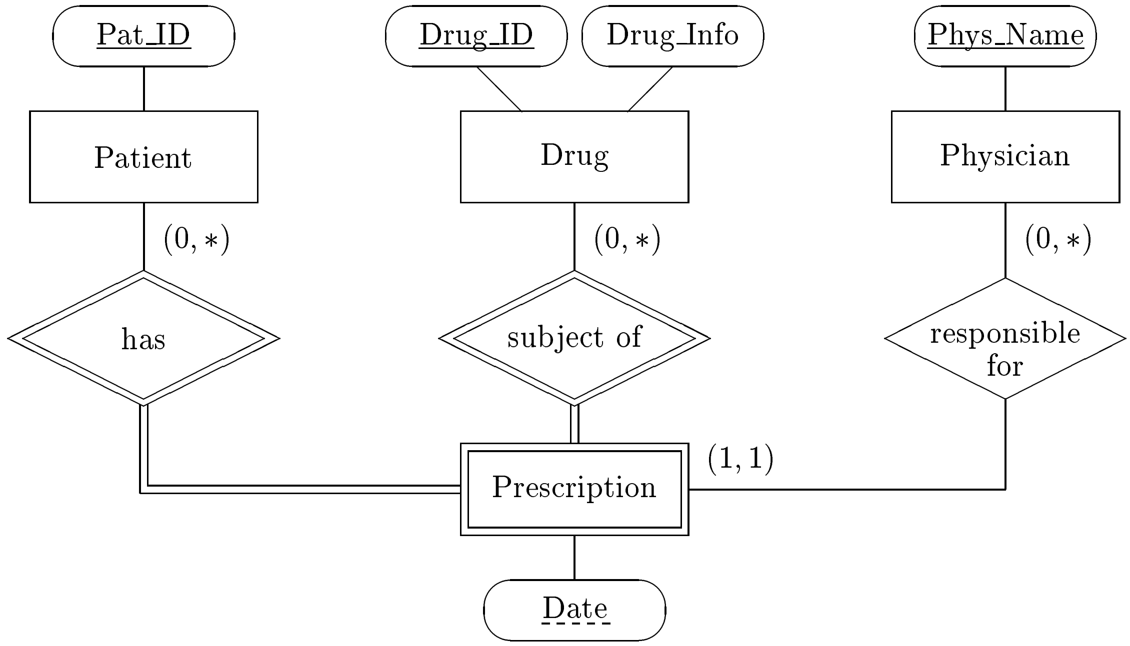
a) This solution has an entity for patients and a weak entity for prescriptions:



☐ Correct
☐ Wrong, Reason: ______________________________

____

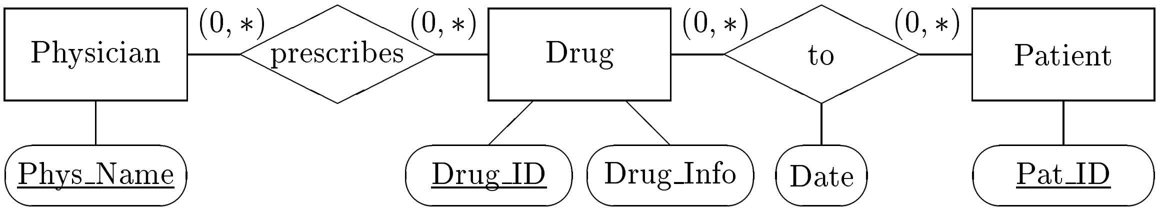
b) This solution models patient, drug and physician as entities and prescription as a weak entity:



☐ Correct
☐ Wrong, Reason: ______________________________

____

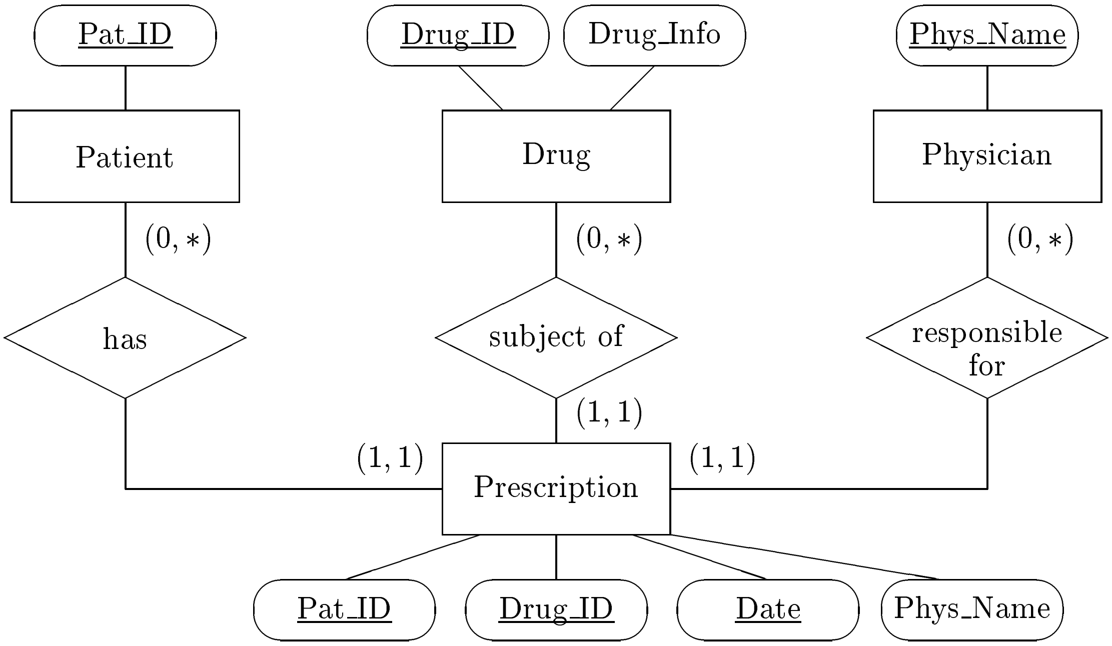
c) This solution puts drug in the middle between physician and patient:



☐ Correct
☐ Wrong, Reason: ______________________________

____

d) This solution is similar to b), but doesn't use a weak entity:



☐ Correct
☐ Wrong, Reason: ______________________________

____

Exercise 2 (ER Design, Oracle Designer Notation) 8 Points

Please develop an Entity-Relationship diagram in Oracle Designer notation for Santa Claus. He needs to store information about:

- Families: The last name of the family and the address (street address, city, state, and ZIP-code). Families are assigned an identifying number (Santa Claus doesn't like this, but he wants to keep track of families which move to another address. So the address cannot be used in a primary key.) Only the current address needs to be stored (no historical information). All attributes are not null. It will not be necessary to store later other information about cities and states.
- Children: Their first name, their birth year, and whether they were very well behaved, normally well behaved, or badly behaved. Children are identified by first name and family. For this exercise, please assume that all children have a family. Santa Claus treats children without a family in a very special and personal way (not via this database).
- Gifts/Toys which the children might wish. They have a name which uniquely identifies them. Some, but not all, have a minimum age below which they might be dangerous.
- Wishes: who (which child) would like what (which toy)? It is also necessary to store which wishes were granted and which not (yet).

Some time after Santa Claus had given these requirements, the analysts come back and ask more questions. He is impressed how thorough they do their job.

- One child can have several wishes, and several wishes can be granted.
- Santa Claus has only families with childs in this database. Families without childs have quite different needs, so he thinks that it is best to develop a separate database (next year).
- It is possible that a child has no wishes.
- It is not necessary to store a quantity (e.g. two toys of the same kind).

If you should feel that constraints are needed which cannot be expressed in the diagram, please write them down as text. E.g. if you have attributes which can take only a few values, you should write this down. You do not have to specify normal data types. Please specify relationship names (they were missing in the solution to the example exam, but they are important.).

There is space for the solution on the next page.

Exercise 3 (Oracle Designer) 8 Points

a) Oracle Designer allows you to define that several attributes should have the same data type. So when you later have to change the data type (e.g. increase the maximal length of a string), you need to change it only at one place and then use a special command for updating all the attributes.

- ☐ True.
- ☐ False.

b) All entities in the global schema (i.e. all entities in the repository) must appear in at least one ER-diagram (not necessarily in the same diagram).

- ☐ True.
- ☐ False.

c) Oracle Designer can create DDL commands only for Oracle Databases.

- ☐ True.
- ☐ False.

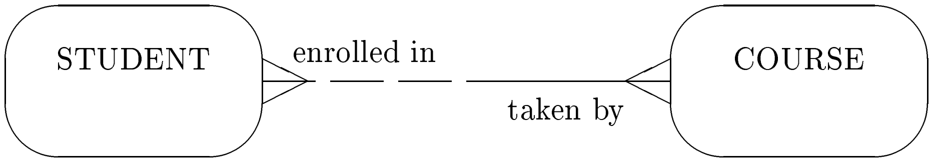
d) Suppose you are not satisfied with the table name the Database Design Transformer creates for a many-to-many relationship. Which tool would you use to change the table name?

- ☐ Entity-Relationship Diagrammer
- ☐ Database Design Transformer
- ☐ Design Editor
- ☐ Repository Reports Utility

e) Oracle designer allows you to define a list of possible values for an attribute (e.g. you can specify that an attribute can have only the values "`Y`" and "`N`").

- ☐ True.
- ☐ False.

f) Consider this relationship:



There are patterns for automatically generated English sentences which describe the cardinalities. Which two sentences would be generated for this relationship?

1. ____

____

2. ____

____

g) Can the relationship in f) be translated into the relational model with the cardinalities enforced by key and foreign key constraints?

- ☐ Yes.
- ☐ No, additional constraints are necessary.

Exercise 4 (FDs, BCNF) 7 Points

Consider the following relation used for storing the result of multiple choice tests like this one:

MULTIPLE_CHOICE_TEST			
STUDENT	QUESTION	ANSWER	CORRECT
John Smith	a	2	Y
John Smith	b	2	N
John Smith	c	3	N
Maria Brown	a	2	Y
Maria Brown	b	3	N
Maria Brown	c	1	Y

We assume that the name uniquely identifies a person. Questions are identified by a letter (`a`, `b`, `c`). Possible answers (options among which the student must choose) are identified by the question and a number (first checkbox, second, etc.). E.g. `2` may be the right answer for question `a`, but the choice `2` for question `b` may be wrong (note that this only an example, it does not necessarily correspond to this test).

Assume that the following function dependencies hold:

- `STUDENT, QUESTION` → `ANSWER`
- `QUESTION, ANSWER` → `CORRECT`

a) What does the functional dependency “`QUESTION, ANSWER` → `CORRECT`” mean? Please check the right explanation (only one is correct):

- ☐ Every question has only one correct answer.
- ☐ The correctness of an answer to a question is independent from the student.
- ☐ It is not possible that the same answer is correct for two different questions.

b) We want to make sure that not more than one answer to a query can be correct. Would the FD “`QUESTION, CORRECT` → `ANSWER`” do the job?

- ☐ Yes. This is the right way to enforce the constraint.
- ☐ This FD is not even satisfied in the given table, but the table satisfies “not more than one correct answer”. The FD is too strong (although it would imply that every question has only one correct answer).
- ☐ The FD is already implied by the given FDs.

c) Can functional dependencies be determined by looking at example data?

- ☐ Yes.
- ☐ Example data can only show that functional dependencies do not hold. You cannot conclude from example data that an FD must hold in general.
- ☐ If an FD holds in an example relation, you can use it for the BCNF test. However, there might be further FDs, which do not hold in this database state, and which also have to be considered.

d) Is the FD "`STUDENT, QUESTION` $\rightarrow$ `CORRECT`" implied by the given two FDs?

- ☐ Yes.
- ☐ No.

e) What is the (minimal) key of the relation with respect to the two given FDs? (It has only one key.)

- ☐ `STUDENT, ANSWER`
- ☐ `QUESTION, ANSWER`
- ☐ `STUDENT`
- ☐ `STUDENT, QUESTION`
- ☐ `STUDENT, QUESTION, ANSWER`

f) Is the relation in BCNF?

- ☐ Yes.
- ☐ No. The FD "`STUDENT, QUESTION` $\rightarrow$ `ANSWER`" violates BCNF.
- ☐ No. The FD "`QUESTION, ANSWER` $\rightarrow$ `CORRECT`" violates BCNF.
- ☐ No. Both FDs violate the BCNF condition.

g) What would be a lossless decomposition (only one answer is correct)?

- ☐ `R1(STUDENT, ANSWER)` and `R2(QUESTION, ANSWER, CORRECT)`
- ☐ `R1(STUDENT, QUESTION)` and `R2(ANSWER, CORRECT)`
- ☐ `R1(STUDENT, QUESTION, ANSWER)` and `R2(QUESTION, ANSWER, CORRECT)`
- ☐ None of the above.