Dr. Stefan Brass
School of Information Sciences
University of Pittsburgh

February 23, 2000

INFSCI 2710 "Database Management" — Midterm Exam —

Instructions

- You have time until 8^{50} (i.e. 2:20) to work on the exercises. The time should be more than sufficient if you know the material well. However, if you have to read longer parts of notes or books, the time will become a problem. So try to solve first the exercises which you can do without using books (or where you know exactly where to look). Don't forget to make at least a guess in the multiple-choice questions.
- Please put the answers into the provided space. If the space is not sufficient, use the back page or attach additional sheets. But mark clearly that there is a continuation, and put your name and the number of the exercise on any additional sheets.
- Please ask if you are not sure that you understand the exercise. Also in real world projects, the designers need to ask the customer again, a single statement of the goals is seldom sufficient.
- This exam has 12 pages. Please check that they are complete.
- Please do not take a copy of this exam with you, also do not take notes about the exercises. You will get the exam back when it is corrected.
- The exam is open book: you can use books or notes, but you cannot use computers.
- Please avoid all behaviour which may be (mis-)understood as cheating. Do not look into the direction of other people's exams, or talk with other students until I have got all exams. It is safest not to sit near your friends.
- Please make sure that I can read your handwriting. Do not use red ink.
- Good luck!

Please Fill In:

Name: ______________________________

Exercise 1 (ER Design) 12 Points

Suppose we want to develop a restaurant guide for Pittsburgh. In order to do this, we have a few professional testers, but we also ask the general public to tell us about their experiences. Of course, all data which we gather must be stored in a database. Please develop an ER-diagram which allows us to store the following information:

- Information about restaurants: What is the name of the restaurant, the street address, the ZIP-code, and the phone number? You can assume that restaurants in Pittsburgh are identified by there name (although this excludes chains).
- For every restaurant, we want to store information about a few dishes the restaurant offers (its specialities). We need the name of the dish and the price. Note that different restaurants can have dishes with the same name.
- Information about test-eaters: What are the first name, the last name, the street address and the ZIP-code of the test eater. You can assume that test-eaters are identified by first and last name (If you think that this is not realistic, you may add other key attributes, but this will not influence your grade.).
- For the subset of professional testers we need in addition the following information: What rate did we agree to pay for every test? (It is probably not very realistic, but we assume a fixed rate, which depends only on the tester, but not on the restaurant tested.) What is his/her phone number? We store phone numbers of professional testers only (not for general public testers).
- For every test, we need the restaurant and the tester, and three grades: One how well the meal tasted, one how good/friendly the service was, and one how nice the room was. In addition, we optionally store a comment.
- Note that the restaurant is tested, not the speciality dishes. We do not store the information which dishes were tasted (except possibly in the comment).
- We do not allow that the same tester tests the same restaurant twice. But of course we try to get tests from different testers for the same restaurant. We have restaurants in our database which were not yet tested, and also information about test eaters, which did not yet submitted a test.

Please define also keys and cardinalities. If other constraints should be needed, it suffices to sketch them in natural language.

There is space for the solution of this exercise provided on the next page.

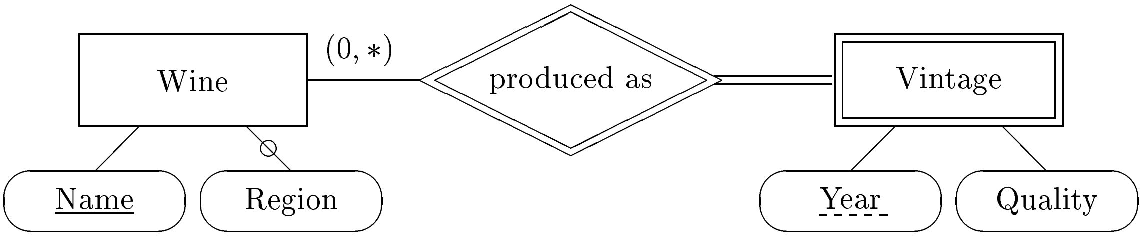
Solution to Exercise 1

Exercise 2 (Comparison of ER-Schemas) 6 Points

Suppose we want to store information about German wines.

- We must distinguish between a wine in general (the brand or vineyard), and the wine produced in a specific year (vintage). E.g. the wine "Oppenheimer Sackträger" is produced in different years. I liked especially "Oppenheimer Sackträger 1994".
- For every wine sort (e.g. "Oppenheimer Sackträger") we store its name, and the region of Germany where the wine was produced (e.g. the vineyard/farm producing "Oppenheimer Sackträger" lies in "Rheinhessen"). Better wines have a defined region, but we also allow to store mixed wines (without a region).
- For every vintage (i.e. a wine sort in a specific year), we want to store the quality of the wine (e.g. "Kabinett", "Spätlese", "Auslese"). "Oppenheimer Sackträger 1994" was a "Auslese". The quality of the same wine produced in different years might be different (depending on the amount of sunshine etc.). However, the region will always remain the same.
- We allow to store information about a wine sort without having information about a specific vintage.

To make the concepts clearer, the following would be a correct schema:

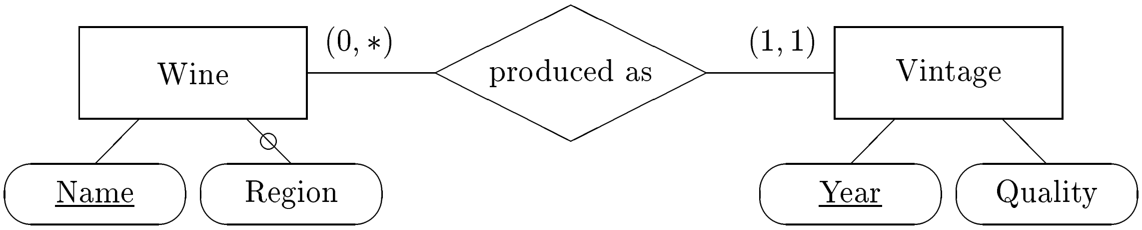


So basically we have to store three facts:

- Wine names.
- The region for some of the wine names.
- The quality, which depends on wine name and year.

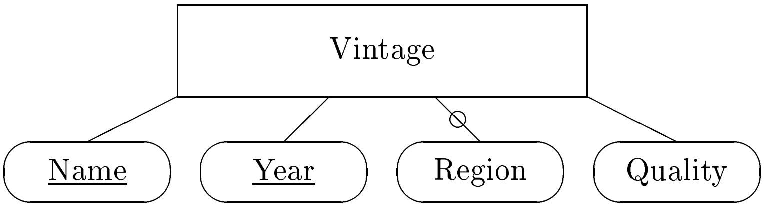
Which of the following ER-schemas are correct (i.e. equivalent to the given schema) and which are not? If a schema is incorrect, please give an explanation. Each subexercise is worth two points. If the schema is wrong, your explanation must show that you mean a real error. Without explanation, or with an incorrect explanation, you will get 0 points for that exercise.

a) This solution avoids the weak entity:



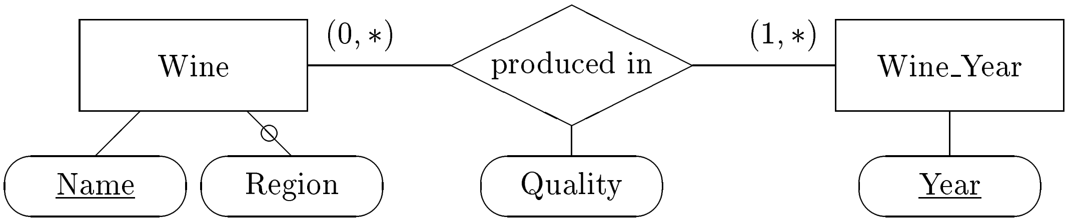
☐ Correct
☐ Wrong, Reason: ______________________________

b) This solution puts all information into a single entity:



☐ Correct
☐ Wrong, Reason: ______________________________

c) This solution makes the year an entity and puts the quality information into a relationship attribute:



☐ Correct
☐ Wrong, Reason: ______________________________

Example Database for Exercises 3 to 8

Consider a company which owns a number of apartment buildings in Pittsburgh. It wants to store information about their buildings, apartments, contracts, and payments:

BUILDINGS				
BNo	BName	Address	ZIP	Built
1	The Chalfont	4742 Centre Ave	15213	1970
2	The Office	135 N. Bellefied Ave.	15260	1965
⋮	⋮	⋮	⋮	⋮

The table "APARTMENTS" contains information about apartments. The column "BNo" is a foreign key referencing "BUILDINGS".

APARTMENTS			
BNo	ANo	Bedrooms	Baths
1	305	2	2
2	722	0	0
2	726	0	0
2	727	0	0
⋮	⋮	⋮	⋮

Next, there is a table with information about contracts. We only store contracts for currently rented apartments in this table (i.e. an apartment is rented if and only if there is a matching row in "CONTRACTS"). The column "Rent" contains the monthly rent, and "Balance" tell us how much the tenant owes us. On the first of every month, "Balance" is increased by "Rent". Whenever we receive a payment, it is subtracted from "Balance". "BNo" and "ANo" in "CONTRACTS" are referencing "APARTMENTS".

CONTRACTS						
CNo	BNo	ANo	Start	Tenant	Rent	Balance
1001	1	305	15-DEC-99	Stefan and Nina Brass	970	0
1002	2	726	01-JUN-00	Stefan Brass	50	100
1003	2	727	01-JAN-99	Michael Spring	70	-70
⋮	⋮	⋮	⋮	⋮	⋮	⋮

"PAYMENTS" contains a record of all payments. "CNo" references "CONTRACTS".

PAYMENTS		
CNo	Date	Amount
1001	29-JAN-00	970
1003	15-JAN-00	70
1003	15-FEB-00	70
⋮	⋮	⋮

All attributes are not null.

Exercise 3 (Keys) 6 Points

Each of the following exercises is worth 2 points. Please check exactly one answer per exercise.

a) Can we declare “BNo” and “ANo” together as an alternative key for “CONTRACTS”? Please check the statement which makes most sense (actually, all except one are really wrong):

- ☐ No, this is impossible. This key is already violated in the given example data.
- ☐ Yes, this means that there can be only one contract per apartment. Since we don’t store historical information, this is ok.
- ☐ No, this “key” would be non-minimal.
- ☐ No, you cannot have two relations with the same key. “ANo” and “BNo” are already key of “APARTMENTS”.

b) Could we declare “ANo” (apartment number) as a key of “APARTMENTS”? (Then “BNo”, the building number, would not be needed to identify apartments.) Please check the statement which makes most sense:

- ☐ No, this is impossible. This key is violated in the given example data.
- ☐ Yes. The given example data clearly show that already the apartment number is sufficient as a key.
- ☐ No, this key is too strong. It excludes database states which can occur in real life. Example data alone are not sufficient to determine a key.

c) Which of the following insert statements leads to a violation of the key “CNo”, “Date” on “PAYMENTS”? Assume that intially only the three rows shown above are in the table. It is possible that more than one statement leads to a violation, in that case check the first statement.

- ☐ INSERT INTO PAYMENTS VALUES (1002, ’29-JAN-00’, 970)
- ☐ INSERT INTO PAYMENTS VALUES (1003, ’15-JAN-00’, 100)
- ☐ INSERT INTO PAYMENTS VALUES (1003, ’20-FEB-00’, 70)
- ☐ None of the above.

Exercise 4 (Foreign Keys) 6 Points

Each of the following exercises is worth 2 points. Please check exactly one answer per exercise.

a) “BNo” in “APARTMENTS” is a foreign key referencing “BUILDINGS”. Does the following hold?

$$\mathsf{APARTMENTS} = \pi_{\mathsf{BNo,ANo,Bedrooms,Baths}}(\mathsf{APARTMENTS} \bowtie \mathsf{BUILDINGS}).$$

- ☐ Yes, this is true.
- ☐ No, the right hand side is only subset of the left hand side.
- ☐ No, the right hand side can be superset of the left hand side.

b) Could we declare “BNo” and “ANo” in “APARTMENTS” as a foreign key referencing “CONTRACTS” (if we defined “BNo” and “ANo” in “CONTRACTS” as an alternate key)? Please check the statement which makes most sense:

- ☐ It is impossible that a key is at the same time a foreign key.
- ☐ It would be possible, but then all apartments would have to be rented (i.e. there must be a contract for them).
- ☐ This foreign key is required if we declare “BNo” and “ANo” as an alternate key in “CONTRACTS”. It is forbidden to declare that key without the foreign key constraint.

c) Consider the foreign key “CNo” in the relation “PAYMENTS” which references “CONTRACTS”. Which of the following operations can never violate the foreign key?

- ☐ Insertion into PAYMENTS
- ☐ Deletion from PAYMENTS
- ☐ Deletion from CONTRACTS
- ☐ All of these operations can violate the foreign key constraint.

Exercise 5 (Relational Algebra) 9 Points

Formulate the following queries in relational algebra. Your queries must work with any database state, not only with the example. You get three points for every correct query. You will lose points for unnecessary complications (e.g. joins which are not needed or projections on all columns). You do not have to care about efficiency (e.g. you will not get more points if you do a selection before the join).

a) Print all information (number, name, address, ZIP-code, and the year of construction) of all buildings built before 1970.

b) Print all payments (contract number, date, amount) for the building with the name "The Chalfont". You cannot assume that you know the building number from the example data (the example data are only an illustration).

c) Print the building name and apartment number of all apartments which are currently not rented, i.e. for which there is no contract.

Exercise 6 (SQL) 9 Points

Formulate these queries in SQL. Note that the queries are not the same as in Exercise 5. Each correct query earns you 3 points.

a) Give all information (contract number, building number, apartment number, start date, tenant name, rent, balance) about contracts where the tenant name ends in "Brass" (e.g. "Stefan Brass", but not "Brass Music Shop").

b) Print the names and balance of all tenants who owe us at least double their monthly rent. Remember that "Balance" in CONTRACTS is the amount owed (if positive).

c) Print building number and apartment number of all apartments with 2 or 3 bedrooms and 2 bathrooms. So the two possibilities are 2 bedroom/2 bathroom apartments and 3 bedroom/2 bathroom appartments.

Exercise 7 (SQL Errors) 6 Points

Which of the following three SQL queries are correct and which are not? It is possible that all queries are correct or all queries are incorrect. If a query is incorrect, please give a short explanation. You will only get points if your explanation shows that you found a real error. Only checking “wrong” will not give you points. I have tried to make sure that every query contains at most one error. But if I have overlooked something, and you tell me about more than one error, you will get extra credit. However, if one of the errors you mention is not a real error, you will lose points. Queries which are syntactically correct but will always give no result tuples count as wrong.

a)
```
SELECT BName, Address
FROM   BUILDINGS
WHERE  ZIP = 15213 OR 15214
```

☐ Correct
☐ Wrong, Reason: ____________________

____

b)
```
SELECT BName
FROM   BUILDINGS
WHERE  ZIP = 15213 AND ZIP = 15214
```

☐ Correct
☐ Wrong, Reason: ____________________

____

c)
```
SELECT Tenant
FROM   Apartments
```

☐ Correct
☐ Wrong, Reason: ____________________

____

Exercise 8 (Relational Algebra Puzzle) — 3 Extra Points

Write a relational algebra expression to compute the following table (i.e. the table should be the result of the query). It is the same as "APARTMENTS", but with the extra column "Rented". "Rented" should be "Y" if there is a contract, and "N" otherwise. Since this exercise is for extra credit, it will be corrected very strictly (partial credit only for nearly correct solutions).

BNo	ANo	Bedrooms	Baths	Rented
1	305	2	2	Y
2	722	0	0	N
2	726	0	0	Y
2	727	0	0	Y
⋮	⋮	⋮	⋮	⋮

Results (To be Filled Out by the Instructor):

Exercise	Points	Max Points
1 (ER Design)		12
2 (Comparison of ER-Schemas)		6
3 (Keys)		6
4 (Foreign Keys)		6
5 (Relational Algebra)		9
6 (SQL)		9
7 (SQL Errors)		6
8 (Relational Algebra Puzzle)		0
Total		54