Uppsala University
Department of Information Technology

Exam 2014-01-13

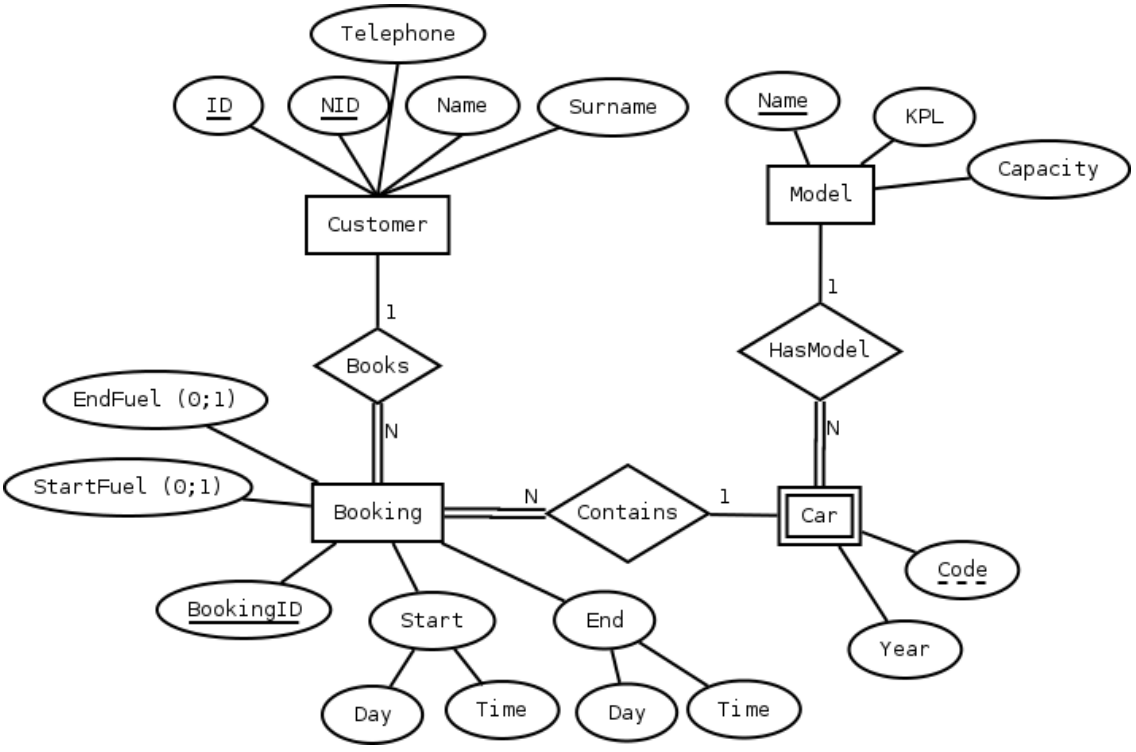
Database Design I (1DL300)

Please notice that for most of the exercises there is not just a single correct solution and we cannot list all the possible alternatives. Therefore, use this document wisely.

1) Questions on theory (12 points)

a) 1
b) 4
c) 2
d) 4

2) ER modeling (16 points)



Business rules:

1) A single booking cannot last longer than 3 days. (but the same car can be booked by the same customer multiple times)

2) The same car cannot be booked more than once during the same time.

Tasks:

a) Design an EER diagram representing this scenario.

b) Clearly indicate all requirements that cannot be represented in the diagram or that you prefer not to represent in the diagram to keep it simple, if any.

3) Translation to the Relational Model (15 points)

One option:

a) Person(Name, ID, IsCoach, Age, IsPlayer, Number).
b) Person.ID
c) None.
d) Age, Number.
e) Number.

Another option:

a) Person(Name, ID)
   Coach(ID, Age)
   Player(ID, Number)
b) The three ID attributes.
c) Coach.ID and Player.ID to Person.ID
d) None.
e) Player.Number.

4) SQL (16 points)

a)
```
SELECT Distinct Age
FROM (Student JOIN Registration ON SID=StudentID) JOIN Course ON CID=CourseID
WHERE Name = 'Database Design 1"
```
b)
```
SELECT SUM(Cost)
FROM Course
WHERE CID LIKE 'IT%'
```
c)
```
CREATE VIEW C_A_AGE(CourseID, A_AGE) AS
SELECT CourseID, AVG(Age)
FROM Student JOIN Registration ON SID=StudentID
GROUP BY CourseID
```
d)
```
SELECT SID
FROM STUDENT
WHERE AGE > any
        (SELECT A_AGE
        FROM Registration NATURAL JOIN C_A_AGE
```

```
        WHERE StudentID = SID)
```

[the query can also be interpreted as: all courses to which the student is registered have a lower average age. In this case, "all" can be used instead of "any"]

or

```
SELECT Distinct SID
FROM (Student JOIN Registration ON SID=StudentID) NATURAL JOIN C_A_AGE
WHERE AGE>A_AGE
```

5) Relational Algebra (5 points)

(a) C(CID, CName) <- Course
S(SID, SName) <- $\pi_{SID, Name}$ (Student)
R(SID, CID) <- Registration
$\sigma_{CName='Database\ Design\ 1'}$ ($\pi_{SName, CName}$ (C $\bowtie$ R $\bowtie$ S))

6) Normalization (14 points)

There are multiple ways to solve this. Here is a compact description of some main options – any of these is fine.

a) CK1: BookingID
CK2: Data, CarID (this may not be indicated as a candidate key)

b) BookingID -> all attributes
Data, CarID -> all attributes (if this has been indicated as a candidate key)
CarID -> CarModel (this is also optional)

IF CarID -> CarModel is not indicated as a FD, then the table is in BCNF and the exercise is completed. Otherwise:

c) If Data, CarID is a CK, the table is in 1NF and not in 2NF. Otherwise the table is in 2NF and not in 3NF.

d) We can change the model of car Cr001 only in row B002, creating an inconsistency.

e) We can remove row B004, losing information on the model of car Cr002.

f) and g)

BOOKING:

Customer	BookingID	Date	CarID
CS001	B001	2014-01-04	Cr001

CS002	B002	2014-01-03	Cr001
CS002	B003	2014-01-12	Cr001
CS002	B004	2014-01-12	Cr002

CAR:

CarID	CarModel
Cr001	Volvo CX60
Cr002	Volvo V40

With a foreign key CAR.CarID -> BOOKING.CarID.

Customer, Date is UNIQUE, is this was marked as a CK (but this was not requested in the exercise).

7) Transactions (16 points)

a) T1, T2 -> set the same price (the highest), then apply a 10% discount to both books.
T2, T1 -> apply a 10% discount to both books, then set the same price (the highest)

b) I represent the two transactions as:
T1: R1(E1), R1(E2), W1(E1 or E2)
T2: W2(E2), W2(E1)

A possible answer is:
R1(E1), R1(E2), W2(E2), W1(E1), W2(E1)
happening when the price of E2 is higher than the price of E1. It is serializable because swapping W2(E2) and W1(E1) in the schedule does not change the result and leads to a serial one.

However, if we want to keep the generic form W1(E1 or E2) it is not possible to write a serializable and non-serial schedule. This is also an accepted answer.

c) For example, R1(E1), R1(E2), W2(E2), W1(E2), W2(E1).
The effect would be to have E1 discounted by 10% and E2 with the original price of E1. However, after all serial schedules the two prices should be the same.

d) T2 would not have write access to E2, because a read lock is hold by T1 at that point.

8) Database terminology (8 points)

Definitions on the textbook.