201300180 Data & Information – Test 2 (1.5 hours)

20 May 2016, 13:45–15:15

Please note:

- Please answer every question on a different sheet of paper (the answers will be distributed to different person for grading).
- You can give your answers in Dutch or English.
- You are not allowed to bring any study materials to the test; essential excerpts from the study materials are available as appendices.

Grade = #points/10

Question 1 (Database Schema) (30 points)

The Marigold Hotel is located in the city, next to the buiness dinstrict. It attracts tourists, who want to visit the city, as well as persons who visit the business district in a professional capacity.
It happens regularly that companies book rooms for their guests or employees. Typically these are company employees from branches in other towns or countries who stay to visit the the company in this city. If a booking is make by a private customer, the customer will pay the bill at the end of the stay. If a booking is made by a company, it is the company who will settle the bill in due course. To that end, some data about the company need to be known. Thi includes a contact person for the company.

Figure 1 shows a class diagram for (part of) the administration of the Marigold Hotel.
Make a database schema that corresponds to this class diagram, using a minimal number of tables.
(*Please note: the database schema is for this class diagram. If the extended class diagram in Question 2 would cause changes to some tables, you do not need to update them.)*

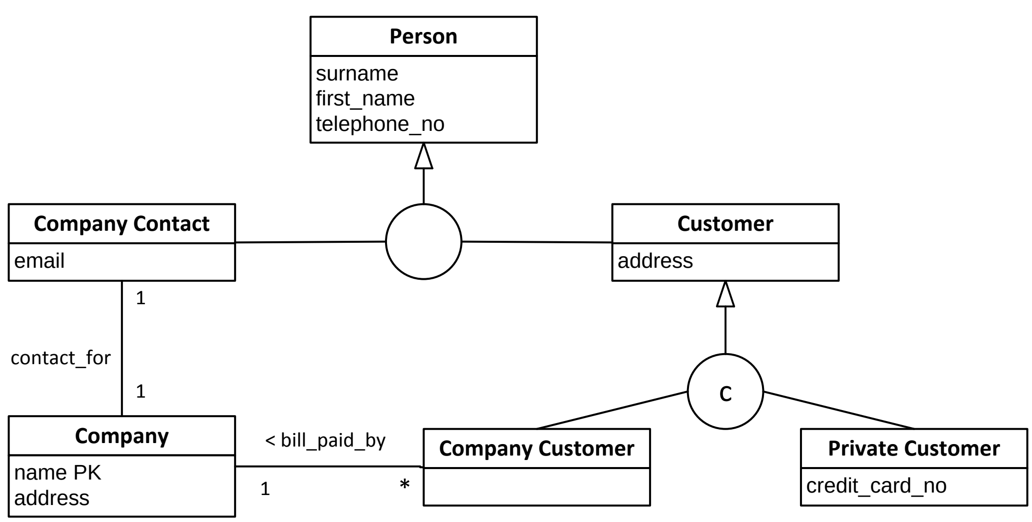


Figure 1: class diagram for part of the Marigold Hotel administration

Question 2 (Class Diagram) (35 points)

Extend the class diagram in Figure 1 with the following information.

The marigold Hotel has single and double rooms with different luxury standards. All rooms with the same outfit have the same price per night. For example: the price per night for a standard double room is € 119. Some rooms may have a better view than others, an extra window, or a few square meters more, but this has no effect on the price.
Occasionally something breaks down in a room. In that case it is possible to indicate in the system that the room won't be available until a particular date. Usually, repairs are done within a day, but in some cases it may take longer.
Last but not least, each room has a unique number.

When a guest (or a company contact person on their behalf) makes a booking, it is registered what room is requested, e.g. a single room "executive style". But a specific room number need not be allocated at the time of booking. When the guest checks in, the receptionist can make a choice among the available rooms – excluding those that are currently in repair.
For each booking the following data are registered: the date at with the booking was made, arrival date, departure date, time of check-in (on the arrival date) and check-out (on the departure date). If the booking is cancelled, the date at which it is cancelled is also recorded. (Note that some of these data will have null values: if the booking is cancelled there is no check-in, and reversed.)
Some companies that regularly make bookings for visitors have a preference for one or more rooms. The preference could be for any reason, e.g.: it is in a quiet corner of the hotel; it is near the parking lot; or simply that company guests usually stay in this room. The reason for a preference is indicated in a text field.

(Private) customers who have not pre-booked a room can make a booking whey they arrive, if room is still available.
When a customer brings an accompanying person (i.e., has booked a double room), the surname, first name, and telephone number of the accompanying person is registered when they check in.
During their stay, customers can make use of various services, which can be paid on the spot or be put on the bill for when they check out. Examples of services are laundry services, drinks at the hotel bar, dinner in the Marigold Restaurant.
If a service is not paid on the spot, the following details will be recorded in the system: description and price of the service, the time at which the service is recorded, as well as the employee who entered it into the system. For the employee is known: surname, first name, telephone number and employee number.

When private customers check out, they pay the bill for the hotel stay and services.
When company customers check out, they pay only the services.
However, to keep the class diagram limited in size, we disregard everything related to payment.

Many customers return to the Marigold Hotel, the next time they visit the city, because of the strategic location and the excellent service.

Question 3 (35 points)

3a) (Functional dependencies) (15 points)

For the administration of a hotel, a relation *R*(*A,B,C,D,E,F,I,R,S*) is defined. For the attributes of *R*, the following holds:

1. *A* is a the arrival date of a booking;
2. *B* is a booking, i.e., a stay in the hotel;
3. *C* is a customer, the person by/for whom a booking is made;
4. *D* is the departure date of a booking;
6. *E* is an employee of the hotel;
5. *F* is a financial institute (i.e., a bank) with which the hotel has an account;
6. *I* is the IBAN number (the account number) of an account with a financial institute;
7. *R* is a room (uniquely identified by a room number);
8. *S* is a service to a specific customer, to be added to the bill;

In addition, the following facts are given:

a. A booking is made for one customer. If the booking is for two persons, one is the customer and the other is an accompanying person.
b. A booking is for one specific room.
c. A booking has a specific arrival date and departure date.
d. A service is recorded by an employee as soon as possible after the service was delivered.
e. A recorded service is added to the bill for one particular stay of one customer.
f. The hotel has multiple accounts with different financial institutions.
g. The IBAN number identifies the account as well as the financial institution

For each of the following – potential – functional dependencies and multivalued dependencies, please indicate whether the dependency holds ("yes") or not ("no").
Give a brief motivation for each answer. Having the right motivation is as important as having the right answer.

i) $AD \longrightarrow B$
ii) $S \longrightarrow AD$
iii) $C \longrightarrow I$
iv) $I \longrightarrow F$
v) $I \twoheadrightarrow F$
vi) $R \twoheadrightarrow IF$
vii) $E \longrightarrow C$
viii) $CE \longrightarrow R$
ix) $CS \longrightarrow R$
x) $B \longrightarrow E$

3b) (Normal forms) (20 points)

Consider the relational schema *R*(*A,B,C,D,R,S,T*) for a hotel administration, where

> *A* is a the arrival date of a booking;
> *B* is a booking;
> *C* is a customer;
> *D* is the departure date of a booking;
> *R* is a room;
> *S* is a service to a specific customer, to be added to the bill;
> *T* is a telephone number of a customer.

with functional dependencies $\mathcal{F}$, defined by

$$\mathcal{F} = \{ B \rightarrow ACDR,\ S \rightarrow B,\ T \rightarrow C \}.$$

1) Which functional dependencies violate the BCNF condition? Why?

2) Apply the algorithm in Appendix 3 to decompose *R* into a set of relational schemas that are all in BCNF. Please give all the details that you need to check in order to verify that you properly executed the algorithm.

3) Are any functional dependencies of *R* lost in the decomposition? If so, which one(s)?

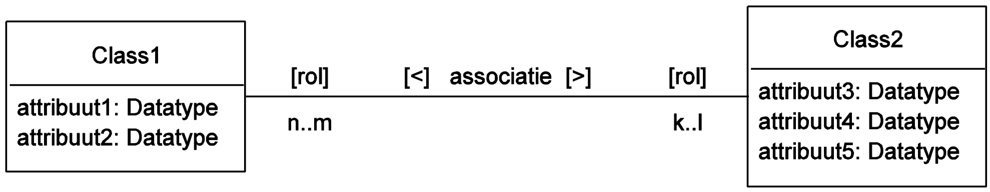
Appendix 1: Notations for class diagrams

meta-notation:

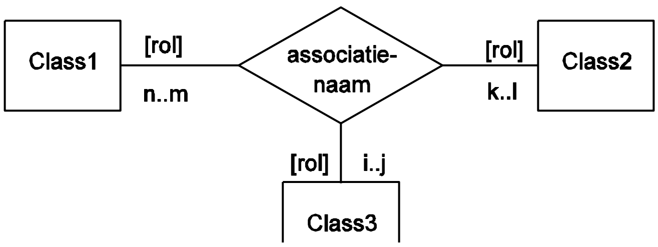
[...]   Optional (can be deleted)

.. | ..   Choice: one of the given alternatives

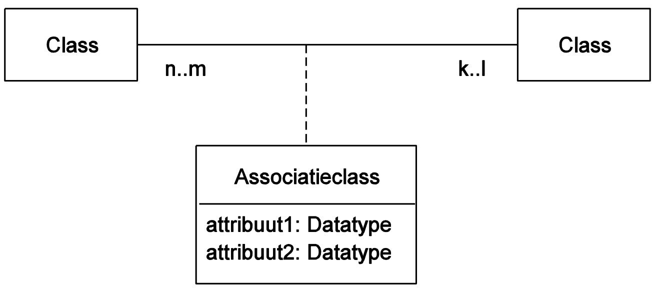
Class and Association



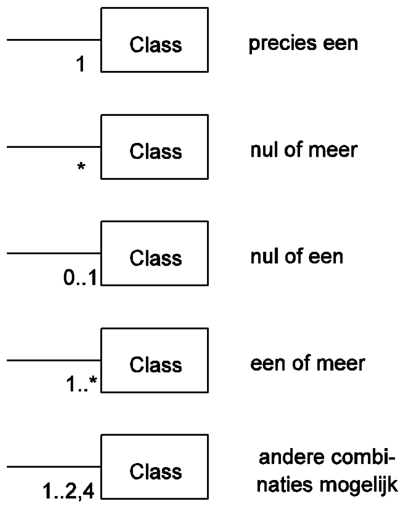
Ternary (or *n*-ary) association



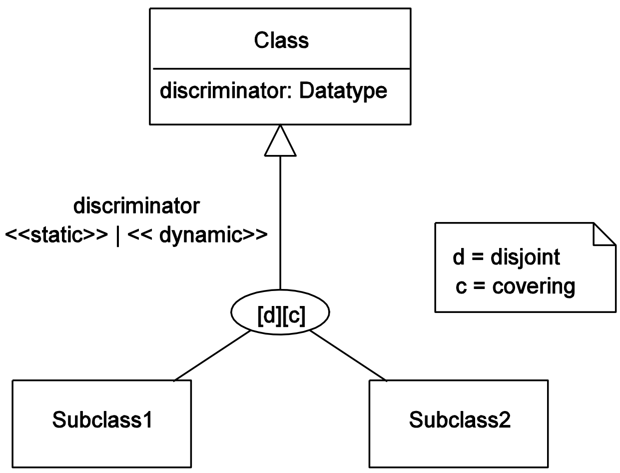
Association class



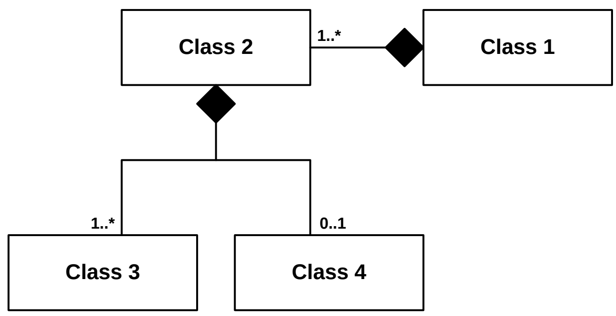
Multiplicity



Generalization



Composition



Appendix 2: Informal syntax for database schema

Informal syntax ('|' for choice and '[ ]' for optional):

```
CREATE TABLE table (
  column [NOT NULL] [UNIQUE] [PRIMARY KEY]
  [ , column ... ]
  [ , PRIMARY KEY (column, ... ) ]
  [ , FOREIGN KEY (column, ... ) REFERENCES table(column, ...)
    [ , FOREIGN KEY ...] ]
  [ , CHECK ( condition ) ]
);

Examples of condition:
   column = value [ (OR | AND) [NOT] column <> value ] |
   column IS [NOT] NULL |
   column [NOT] IN (value, ...) |
   ...
```

Appendix 3: Losless BCNF decomposition algorithm

Definition of BCNF:
A relational schema is in BCNF if for every nontrivial functional dependency the left-hand side is a superkey.

Decomposition algorithm:
Let R be a relational schema with a set of functional dependencies $\mathfrak{F}$.
Let $X \rightarrow Y$ be a functional dependency in $\mathfrak{F}$ which violates the BCNF constraint.

- Decompose R into
  - $R_1(X^+)$
  - $R_2(Z)$ with Z = {X} U {attributes of R not in X^+).
- For i = 1,2:
  - determine $\mathfrak{F}_i$ for R_i by restricting $\mathfrak{F}^+$ to functional dependencies within R_i
  - if R_i is not in BCNF, recursively apply the algorithm