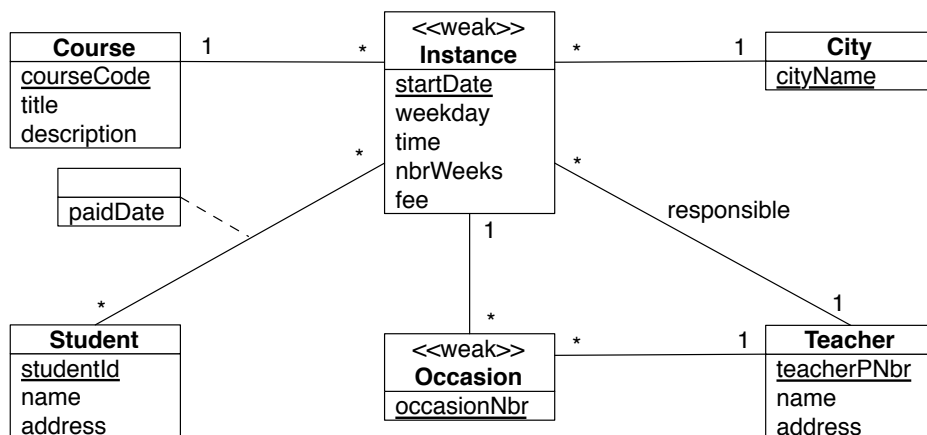


Solutions, Database Technology Examination

2014-08-22

1. E/R diagram:



The weak entity set Instance has been given the key instanceId (instead of courseCode + cityName + startDate). occasionNbr is the ordinal number of an occasion of a course instance (1, 2, 3, ...).

Courses(courseCode, title, description)

Cities(cityName)

CourseInstances(instanceId, courseCode, cityName, startDate, weekday, time, nbrWeeks, fee, responsibleTeacherPNbr)

Students(studentId, name, address)

StudentsTakeCourses(*studentId*, *instanceId*, paidDate)

Teachers(teacherPNbr, name, address)

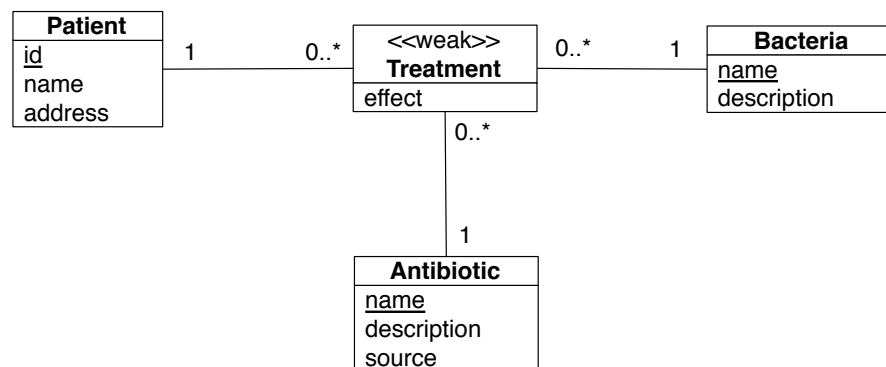
Occasions(*instanceId*, occasionNbr, *actualTeacherPNbr*)

There are no other functional dependencies except for the key dependencies, so the relations are in BCNF.

Total revenue per course for courses in Lund during 2014:

```
select courseCode, title, sum(fee)
from CoursesInstances natural join StudentsTakeCourses
where cityName = 'Lund' and startDate like '2014%' and paidDate is not null
group by courseCode, title
order by courseCode;
```

2. E/R diagram:



```

create table Treatments (
    patientId int,
    bacteriaName varchar(30),
    antibioticName varchar(30),
    effect int check (effect >= 0 and effect <= 10),
    primary key (patientId, bacteriaName, antibioticName),
    foreign key (patientId) references Patients(id),
    foreign key (bacteriaName) references Bacteria(name),
    foreign key (antibioticName) references Antibiotics(name)
);

```

```

select name
from Antibiotics
where name not in (select antibioticName from Treatments);

```

```

select count(distinct patientId)
from Treatments
where bacteriaName = 'S. aureus';

```

```

select id, name, address
from Patients
where id in (select patientId
            from Treatments
            group by patientId
            having count(*) > 1);

```

```

select antibioticName, source
from Antibiotics, Treatments
where bacteriaName = 'S. aureus'
      and Antibiotics.name = antibioticName
group by antibioticName, source
having min(effect) >= 7;

```

3. We have the following functional dependencies:

- FD1. $A \rightarrow B$
 FD2. $BC \rightarrow E$
 FD3. $DE \rightarrow A$

a) The attributes CD are not on the right-hand side of any FD, so they must be part of all keys.
 Closure of these attributes:

$$\{CD\}^+ \Rightarrow \{CD\}$$

$\{CD\}$ isn't a key. Three-attribute subsets which include CD:

$$\begin{aligned} \{ACD\}^+ &\Rightarrow \{ACD\} \xrightarrow{FD1} \{ACDB\} \xrightarrow{FD2} \{ACDBE\} \\ \{BCD\}^+ &\Rightarrow \{BCD\} \xrightarrow{FD2} \{BCDE\} \xrightarrow{FD3} \{BCDEA\} \\ \{CDE\}^+ &\Rightarrow \{CDE\} \xrightarrow{FD3} \{CDEA\} \xrightarrow{FD1} \{CDEAB\} \end{aligned}$$

$\{ACD\}$, $\{BCD\}$, and $\{CDE\}$ are keys. There are no larger subsets that do not contain these keys, so we have found all keys.

The relation is not in BCNF since at least one (in fact all) of the FD's violates the BCNF condition, i.e. the left-hand side of the dependency isn't a superkey. It is in 3NF since the right-hand sides of the dependencies (B, E, and A) are part of a key.

b) Decompose starting from FD1, $A \rightarrow B$:

- R1(A, B) in BCNF. A is the key, no other FD's.
 R2(A, C, D, E) not in BCNF. ACD and CDE are keys and $DE \rightarrow A$.

Decompose R2 using $DE \rightarrow A$:

- R2a(A, D, E) in BCNF. DE is the key, no other FD's.
 R2b(C, D, E) in BCNF. No FD's.

4. The index is a data structure which the DBMS uses to speed up queries on the *name* attribute (advantage). The index must be updated when the table is updated, which decreases performance. Also, the index takes up space (disadvantages).

5. "Pushing selection" means that the where constraints in an SQL query should be tested as early as possible. This results in smaller relations, which means that there will be less work for the DBMS in the rest of the query.

6. Errors:

- The file contains two <dict> elements, but the top level entity must be a plistObject,
- the second <dict> tag has no corresponding </dict>,
- the tag <year> should be <integer> (presumably).