DNF Data Model C

Relationalisation Eliminates Theory

Identification

Domination Normal Form - Decomposing Relational Database Schemas (sic)

Henning Köhler, Massey University, Palmerston North, New Zealand.

5 An Example

A (sic) university has oral examinations at the end of each semester, and wants to manage related data using a relational database. The relevant attributes to be stored are

$R = \{Student, Course, Chapter, Time, Room\}$

Here Chapter denotes a chapter from the course textbook the student will be examined about. Every student can get examined about multiple chapters, and chapters may vary for each student. Multiple students can get examined at the same time in the same room, but the course must be the same. Further constraints are that a student gets examined for a course only once, and can't be in multiple rooms at the same time.

Errata

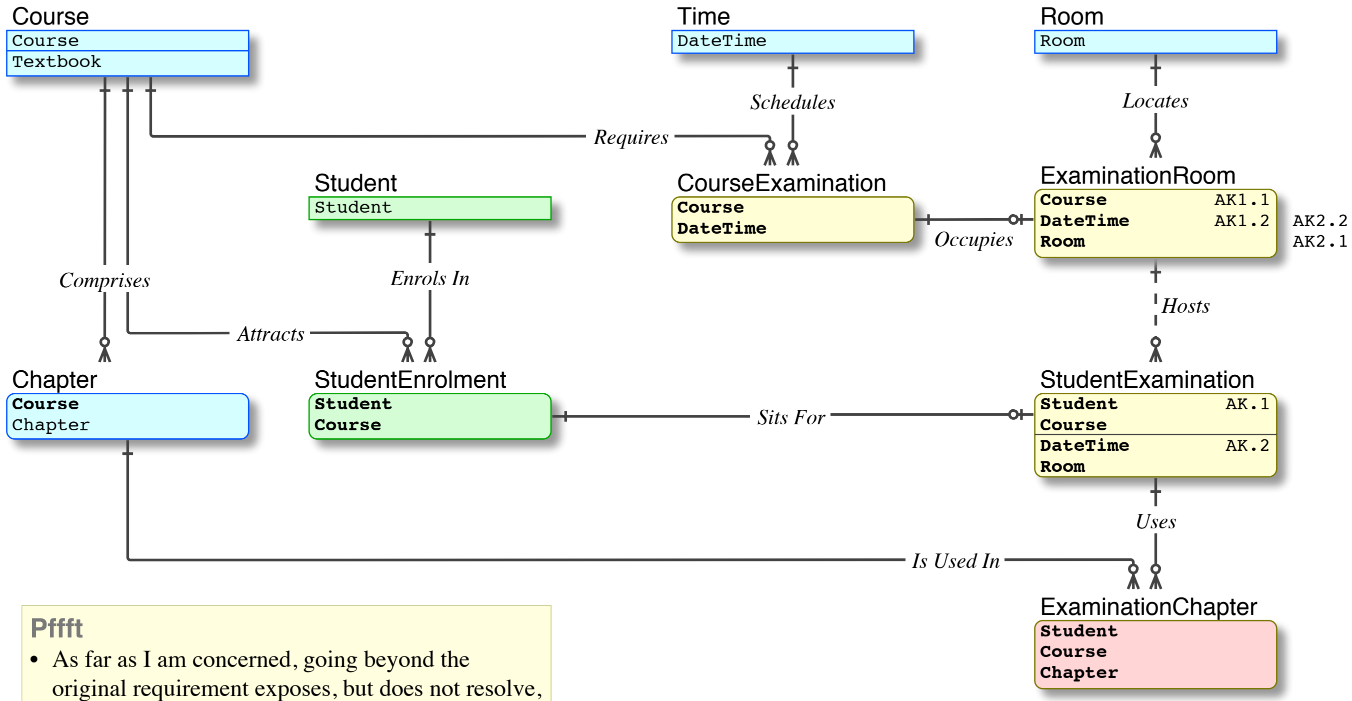
The gentleman expresses himself backwards. Second, some requirements are missing. Based on the possibility that the instance table given on page 6 is the universal relation, by "multiple chapters" apparently he means:

- multiple courses per student
- one exam per course
- multiple chapters per exam, in one sitting.

DNF Data Model C
Examination

This presents a model for the requirement in Köhler's DNF paper, **plus** Nicola's interpretation (for bonus marks), for **evaluation purposes only**:

- 'one predicate is “An exam for Course is scheduled at DateTime” (which may be populated, say, while the course has not finished yet)
- the other is “The exam session for Course scheduled on DateTime will take place in Room” (which may be populated, say, a few days before the exam session)'.



Pffft

- As far as I am concerned, going beyond the original requirement exposes, but does not resolve, numerous issues, and adds complications.
- the creature is ugly as sin, it should be put out of its misery before it harms the eye.

IDEF1X Notation

Predicate

Data modellers and developers can *read* all the Predicates directly from the data model. However, users and Relational novices cannot, thus it is provided for them. These are *all* the Predicates (Constraints; Business Rules) that govern the data, expressed in formal syntax. (For the sake of brevity, two Predicates with the same two variables can be combined with *and*).

```
Chapter is Dependent on, and is an element of, 1 Course
    Chapter is Identified by ( Course, Chapter )
    Chapter Identifies, and is used in 0-n ExaminationChapters
Course is Independent
    Course is Identified by ( Course )
    Course is Described by ( Textbook )
    Course Identifies, and requires  0-n CourseExaminations
    Course Identifies, and attracts 0-n StudentEnrolments
    Course Identifies, and comprises 0-n Chapters
CourseExamination is Dependent on, and a requirement of, 1 Course
    CourseExamination is Dependent on, and a schedule of, 1 Time
    CourseExamination is Identified by( Course, DateTime )
    CourseExamination occupies 0-1 ExaminationRoom
ExaminationChapter is Dependent on, and used in, 1 StudentExamination
    ExaminationChapter is Dependent on, and uses, 1 Chapter
    ExaminationChapter is Identified by ( Student, Course, Chapter )
ExaminationRoom is Dependent on, and occupied by, 1 CourseExamination
    ExaminationRoom is Dependent on, and located in, 1 Room
    ExaminationRoom is Primarily Identified by   ( Course, DateTime, Room ) -- establish PK for migration
    ExaminationRoom is Alternately Identified by ( Course, DateTime )       -- preserve 1::0-1 relation
    ExaminationRoom is Alternately Identified by ( DateTime, Room )         -- prevent double booking
    ExaminationRoom hosts 0-n StudentExaminations
Room is Independent
    Room is Identified by ( Room )
    Room Identifies, and locates 0-n ExaminationRooms
Student is Independent
    Student is Identified by ( Student )
    Student Identifies, and enrols in 0-n StudentEnrolments
StudentEnrolment is Dependent on, and an enrollment of, 1 Student
    StudentEnrolment is Dependent on, and attracted by 1 Course
    StudentEnrolment is Identified by ( Student, Course )
    StudentEnrolment Identifies, and sits for 0-1 StudentExamination
StudentExamination is Dependent on, and a sitting of, 1 StudentEnrolment
    StudentExamination is Primarily Identified by   ( Student, Course )
    StudentExamination is Alternately Identified by ( Student, DateTime )
    StudentExamination is hosted by 1 ExaminationRoom
    StudentExamination Identifies, and uses, 0-n ExaminationChapter
Time is Independent
    Time is Identified by ( DateTime )
    Time Identifies, and schedules, 0-n CourseExaminations
```