School of Computer Science, McGill University

COMP-421 Database Systems, Winter 2025[*]

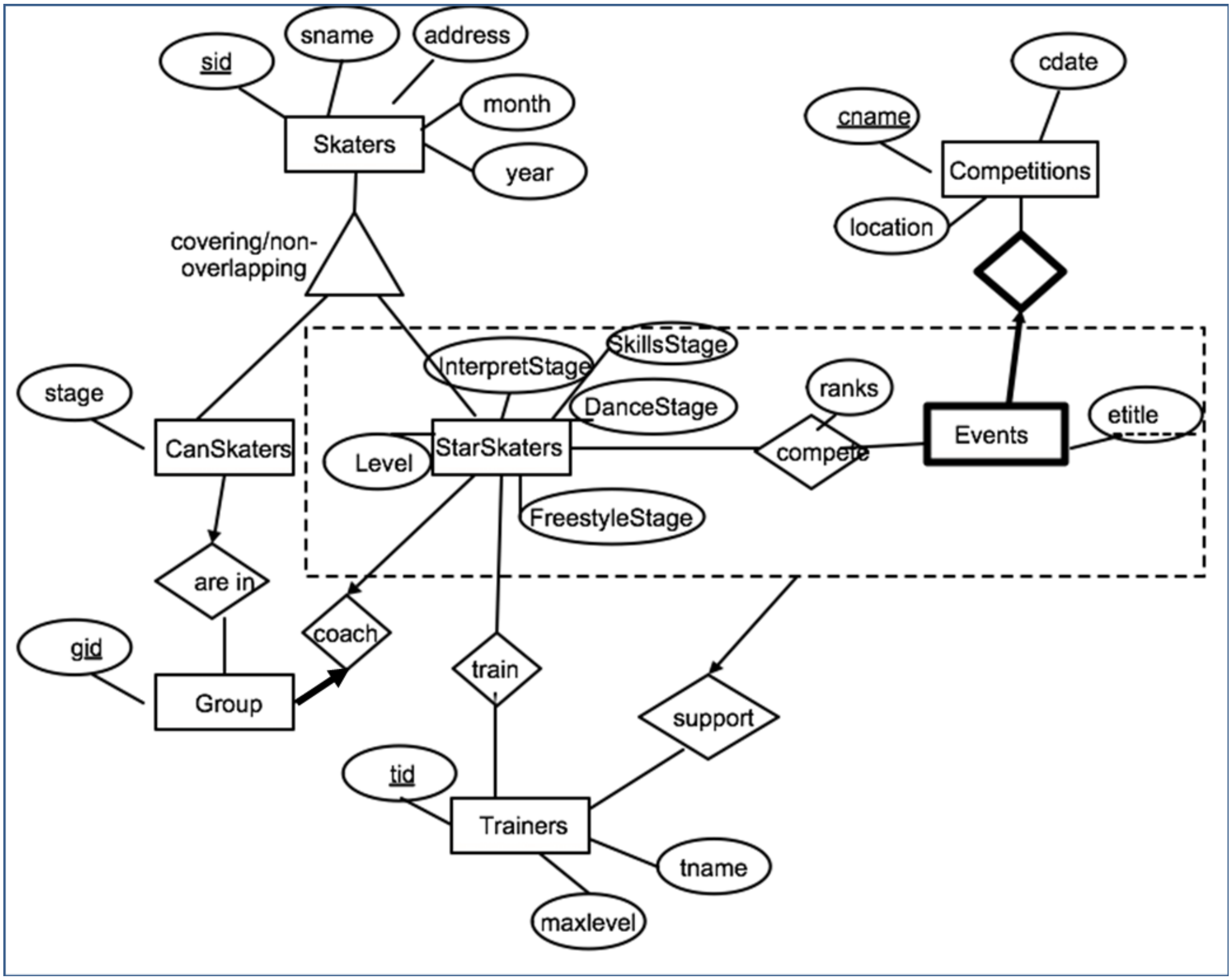
Written Assignment 1: Data Modeling

Due date: January 29, NOON

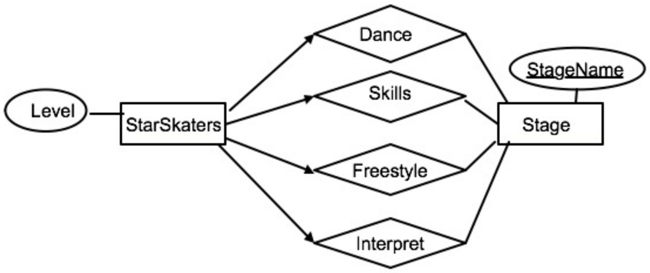
A figure skating club keeps track of its skaters, trainers and the competitions that skaters participate in. The information presented here is taken partially from the SkateCanada webpage. For sake of this exercise, the programs described here are considerably simplified compared to the real SkateCanada programs.

1. (75 Points) Design the E/R diagram for the following specification. Additionally, indicate in a subsection written in textual format any constraints that you might not be able to express in the E/R diagram. For each entity set, decide on the primary keys. If you think there is no appropriate primary key, you can introduce an artificial key attribute.
   - The system keeps track of all skaters of the club. Each skater has a unique id, a name, an address. The month and year of birth are also recorded.
   - Skaters can be either in the CanSkate (beginners) or in the StarSkate program (advanced).
   - The level of CanSkate skaters is defined by the *stage* they have achieved. Stage 1 is the most basic stage (you basically now how to "walk" in forward direction), and Stage 7 the most advanced (simple jumps and pirouettes...). For each skater the club keeps track of the stage they have achieved.
   - CanSkate skaters practice their skills within groups, i.e., each skater is assigned to a CanSkate group.
   - When a CanSkate skater has achieved stage 7, the next term they can join the StarSkate skaters. StarSkate skaters can perform tests in the categories *Skills, Dance, Freestyle, Interpretation.* In each of these categories, depending on the tests they have passed, the skater is at stage *preliminary, bronze, silver, gold, diamond.* The club does not keep track of the individual tests the skaters have passed, but it keeps track of the stage a skater has achieved for each of the categories.
   - StarSkate skaters are grouped depending on the stages they are in for each of the categories. As there is so much variety possible, the grouping is quite course-grained and simply distinguishes between the levels *Junior, Intermediate, Senior.* Thus, the club keeps track of the level of each StarSkate skater.

- StarSkate skaters can volunteer to coach CanSkate skaters. More precisely a StarSkate skater can be assigned to a group of CanSkate skaters as a coach. Each coach can coach at most one group, and each group is coached by exactly one StarSkate skater.
- The club has several professional trainers for the StarSkate skaters. The database should store the names of these trainers. Furthermore, each trainer has a maximum level (Junior, Intermediate, Senior), they are allowed to train. That is, trainers can only train StarSkate skaters that are at a level lower or equal to this maximum level.
- A StarSkate skater can be assigned several trainers and a trainer can train several skaters.
- StarSkate skaters can compete in competitions. Competitions have a name, and have a date and a location.
- Each competition is split into many events. An event has a title (e.g., Junior-fem-Gr2) that is unique within the titles of that particular competition. Typically around 6 to 10 skaters participate in each of these events, and a skater can participate in several events of a given competition. That is, a StarSkate skater of our club is not actually assigned to a competition but rather to individual events of such a competition. The participants of an event are ranked by judges depending on their performance. (That is, if there are 6 skaters, then there will be ranks 1 to 6). The club wants to keep track of the rank each of their skaters has achieved in the events they have participated.
- For each event a skater participates in, they are assigned one trainer who accompanies them to the ice rink and provides advice. It's usually one of the skater's personnel trainers but this is not necessarily always true.



The possible values for SkillStage, DanceStage, FreestyleStage and InterpretStage would be *preliminary, bronze, silver, gold, diamond.*
Alternatively, these stages for the different categories could also be represented as an own entity set, and each skater would have 4 relationship sets representing the different categories. See the schema below:



Constraints:

- We cannot describe that for StarSkaters the stage values are predefined and the only possible values are *preliminary, bronze, silver, gold, diamond.*
- For CanSkaters we cannot represent that stage only goes from 1 to 7.
- We cannot express that in each event there are at most 10 skaters.
- We cannot express that the ranks within an event must all be disjoint and equal to the number of participants.

2. (25 Points) Translate the diagram from 1) into a relational schema. Write the tables in the form R(A1,...An). Indicate primary keys (by underlining them) and foreign keys by indicating to which table/attribute they refer. Indicate if attributes should be NOT NULL (no need to do this for primary keys, they are always NOT NULL). Indicate in a subsection written in textual format if there are other constraints depicted in the ER schema that you cannot (yet) describe within the relational model.

   - Skaters(sid, sname, address, month, year)
   - CanSkaters(sid, stage, gid)
     sid FK to Skaters, gid FK to Group
   - StarSkaters(sid, level, interpretstage, skillstage, dancestage, freestylestage)
     sid FK to Skaters
   - Group(gid, sid)
     sid FK to StarSkaters and sid unique and NOT NULL
   - Trainers(tid, tname, maxlevel)
   - Train(tid, sid
     sid FK to StarSkaters
     tid FK to Trainers
   - Competitions(cname, cdate, location)
   - Events(cname, etitle)
     cid FK to Competitions
   - Compete(sid, cname, etitle, rank, tid)
     sid FK to SarSkaters
     cname, etitle FK to Events
     tid FK to Trainers

   In principle, if a ISA hierarchy is covering/non-overlapping I could skip the superclass (skaters) and only have the subclasses (CanSkaters and StarSkaters) with all attributes of Skaters. The problem is that with this there could be CanSkaters and StarSkaters with the same sid. When now a CanSkaters moves to be a StarSkater somebody else might already have their sid, and they would need to be assigned a new one. Therefore, I left it separated. But with this, the expression covering/non-overlapping cannot be enforced. Note that both translations are correct.