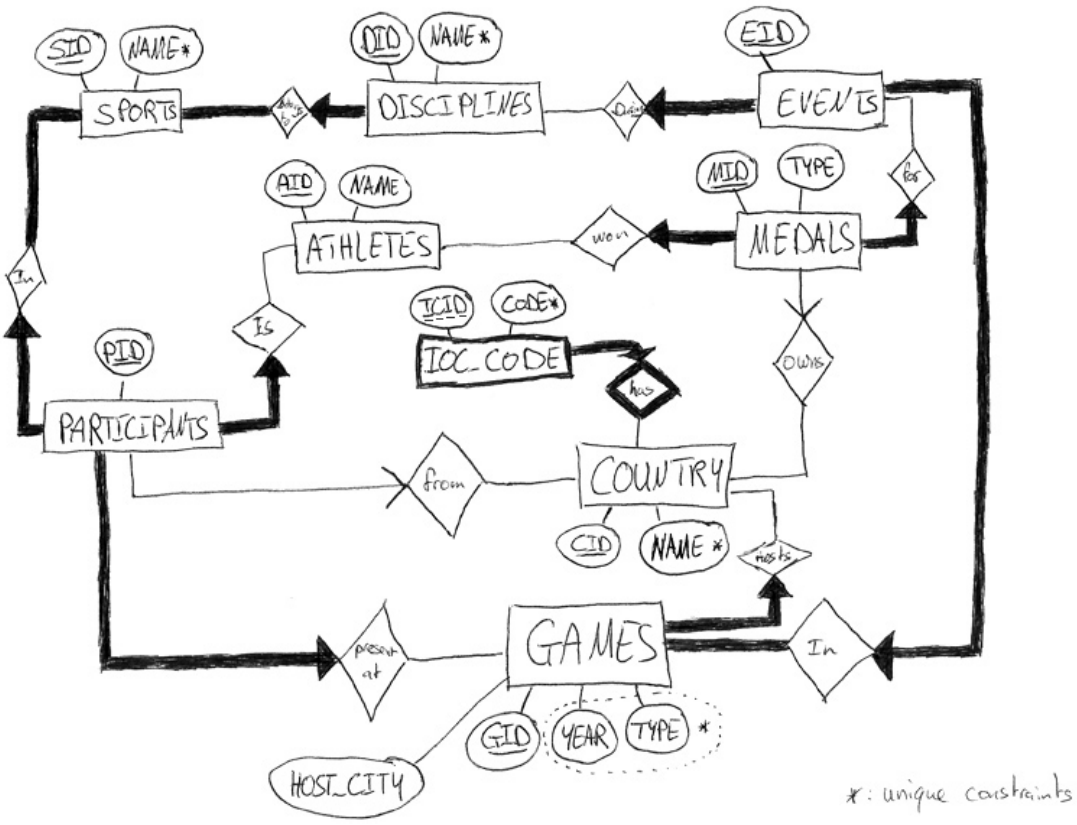
Database project - deliverable 1

Arthur Giroux 205443
Colla Rensch 205814
Valentin Matter 203447

March $24th$ 2013

Deliverable 1

1 ER model



---

2 Tables creation

```
1  CREATE TABLE COUNTRIES
2  (
3    CID INTEGER NOT NULL
4  , NAME VARCHAR2(50) NOT NULL
5  , PRIMARY KEY ( CID )
6  , CONSTRAINT unique_country_name UNIQUE ( NAME )
7  );
```

```
8
9  CREATE TABLE IOC_CODE
10 (
11   ICID INTEGER NOT NULL
12 , CODE CHAR(3) NOT NULL
13 , CID INTEGER NOT NULL
14 , PRIMARY KEY ( ICID )
15 , FOREIGN KEY ( CID ) REFERENCES COUNTRIES( CID ) ON DELETE CASCADE
16 , CONSTRAINT unique_ioc_code_name UNIQUE ( CODE )
17 );
18
19 CREATE TABLE SPORTS
20 (
21   SID INTEGER NOT NULL
22 , NAME VARCHAR2(100) NOT NULL
23 , PRIMARY KEY ( SID )
24 , CONSTRAINT unique_sport_name UNIQUE ( NAME )
25 );
26
27 CREATE TABLE ATHLETES
28 (
29   AID INTEGER NOT NULL
30 , NAME VARCHAR2(200) NOT NULL
31 , PRIMARY KEY ( AID )
32 );
33
34 CREATE TABLE DISCIPLINES
35 (
36   DID INTEGER NOT NULL
37 , NAME VARCHAR2(200) NOT NULL
38 , SID INTEGER NOT NULL
39 , PRIMARY KEY ( DID )
40 , FOREIGN KEY ( SID ) REFERENCES SPORTS( SID )
41 , CONSTRAINT unique_discipline UNIQUE ( NAME, SID )
42 );
43
44 CREATE TABLE GAMES
45 (
46   GID INTEGER NOT NULL
47 , YEAR INTEGER NOT NULL
48 , TYPE VARCHAR2(50) NOT NULL
49 , HOST_CITY VARCHAR2(200) NOT NULL
50 , CID INTEGER NOT NULL
51 , PRIMARY KEY ( GID )
52 , FOREIGN KEY ( CID ) REFERENCES COUNTRIES( CID )
53 , CONSTRAINT unique_game UNIQUE ( YEAR, TYPE )
54 );
55
56 CREATE TABLE EVENTS
57 (
58   EID INTEGER NOT NULL
59 , GID INTEGER NOT NULL
60 , DID INTEGER NOT NULL
61 , PRIMARY KEY ( EID )
62 , FOREIGN KEY ( GID ) REFERENCES GAMES( GID )
63 , FOREIGN KEY ( DID ) REFERENCES DISCIPLINES( DID )
64 , CONSTRAINT unique_event UNIQUE ( GID, DID )
65 );
66
67 CREATE TABLE PARTICIPANTS
68 (
69   PID INTEGER NOT NULL
70 , AID INTEGER NOT NULL
71 , CID INTEGER
72 , GID INTEGER NOT NULL
73 , SID INTEGER NOT NULL
74 , PRIMARY KEY ( PID )
75 , FOREIGN KEY ( AID ) REFERENCES ATHLETES(AID)
76 , FOREIGN KEY ( CID ) REFERENCES COUNTRIES(CID)
```

```
77  , FOREIGN KEY ( GID ) REFERENCES GAMES(GID)
78  , FOREIGN KEY ( SID ) REFERENCES SPORTS(SID)
79  , CONSTRAINT unique_participant UNIQUE ( AID, CID, GID, SID )
80  );
81
82  CREATE TABLE MEDALS
83  (
84    MID INTEGER NOT NULL
85  , TYPE VARCHAR2(50) NOT NULL
86  , CID INTEGER
87  , EID INTEGER NOT NULL
88  , AID INTEGER NOT NULL
89  , PRIMARY KEY ( MID )
90  , FOREIGN KEY ( CID ) REFERENCES COUNTRIES(CID)
91  , FOREIGN KEY ( AID ) REFERENCES ATHLETES(AID)
92  , FOREIGN KEY ( EID ) REFERENCES EVENTS(EID)
93  , CONSTRAINT unique_medalist UNIQUE ( CID, EID, AID )
94  );
```

---

3 Remarks

- Athletes is the entity that stores the information of an athlete who can then participate in multiple sports or games. Which means that if an Athlete competes twice he will have only one entry in the ATHLETES table but two in the PARTICIPANTS table.
- Each game should at least have one event otherwise nothing happened during he games. The same applies to sports, a sport must have at least one participant, otherwise the sport never took place during any games.
- Countries, sports and disciplines must have a unique name as the opposite would have no sense. For games it is the pair (YEAR,TYPE) that must be unique.
- Due to problem with their federations some athletes may present themselves without representing a country.
- Some medals could not be associated to a country as some athletes aren't as for the above point.
- The numbers of countries, athletes and events for a given game isn't stored in the GAMES entity as they can be easily computed using some COUNT query.
- The names of the events, disciplines and games are not stored as they are only the concatenation of information stored in other tables.

Deliverable 2

Modifications

We had to slightly modify our model as in forced each athlete to have a binding participant, but we noticed that there is a quite big amount of them that do not and therefor we decided to slightly change our code as to allow athletes to not have a binding participant.

Data import

We chose to import the data using Java. We decided that any data that would generate a non-existent foreign key would be dropped was would any inconsistent incomplete data (i.e. medals

without color)

Data import was made using Java. When a foreign key couldn't be found in the foreign table or when data was incomplete (medals without color, for instance), we dropped the data.

Queries

A) Simple query using multiple ANDs

B) We have an outer-query which seeks gold medalist in sports that appear in the nested query, which computes all sports that have appeared only once

C) We retrieve the minimum year in which each country won its first medal using a subquery and then use a simple query to get the place hosting the corresponding games.

D) We unite (UNION) two same queries using "Summer" for one and "Winter" for the other and compute the number of medals for each country given the type (Summer or Winter), we then order them despondingly and limit the table to 1

E) Simple GROUP BY + HAVING query

F) We use two table of participants and two tables for countries and then just use ANDs to make find the athletes that competed for at least two countries

G)

H) We simply take the COUNTRIES and use a nested query to delete all entries that do not appear in the MEDALS table

Front-end

Our web front-end is available at `http://db.tamere.ch/`. It was made using PHP and MySQL.