

Projet base de données

Records en Rubik's Cube

Notre projet vise à tirer des statistiques et des informations les records de Rubik's Cube (3x3x3 ou autre) à partir de fichiers TSV téléchargeables sur le site de la World Cube Association (WCA : <https://www.worldcubeassociation.org/results/misc/export.html>). Le site propose également un export en SQL, mais nous ne nous en servons pas.

Nous cherchons à extraire les informations suivantes :

- A quel round a eu lieu le record ? (\Rightarrow est-ce que ça arrive plus souvent en finale ou en poules ?)
- Qui a fait ce record? Où et quand ?
- Quelle est la répartition des records :
 - temporellement
 - géographiquement (continent ou pays)
- etc ...

Nous allons extraire seulement les informations qui nous sont utiles des fichiers TSV.

Nous nous retrouvons avec un modèle à 10 entités, et 11 associations. Pour des raisons qu'on ignore, JMerise nous bloque à 15 blocs (entité ou association) maximum, alors qu'on nous avait dit en cours que le maximum était à 50 :(.

Pseudo-code :

ENTITES :

Result(resultId, competitionId, eventId, roundTypeId, best, average, personId, value1, value2, value3, value4, value5, recordSingleType, recordAverageType

PRIMARY KEY resultId

FOREIGN KEY competitionId ON Competition(competitionId)

FOREIGN KEY eventId ON Event(eventId)

FOREIGN KEY roundTypeId ON RoundType(roundTypeId)

FOREIGN KEY personId ON Person(personId))

Competition(competitionId, competitionName, cityId, dateBeginId, dateEndId, venue

PRIMARY KEY competitionId

FOREIGN KEY cityId ON City(cityId)

FOREIGN KEY dateBeginId ON Date(dateId)

FOREIGN KEY dateEndId ON Date(dateId))

Date(dateId, date

PRIMARY KEY dateId)

City(cityId, cityName, countryId, cityLongitude, cityLatitude

PRIMARY KEY cityId

FOREIGN KEY countryId ON Country(countryId))

Country(countryId, countryName, continentId, iso2

PRIMARY KEY countryId

FOREIGN KEY continentId ON Continent(continentId))

Continent(continentId, recordName, continentName

PRIMARY KEY continentId)

Championship(championshipId, competitionId, championshipType

PRIMARY KEY championshipId

FOREIGN KEY competitionId ON Competition(competitionId))

Person(personId, personName, personSurname, countryId, gender

PRIMARY KEY personId

FOREIGN KEY countryId ON Country(countryId))

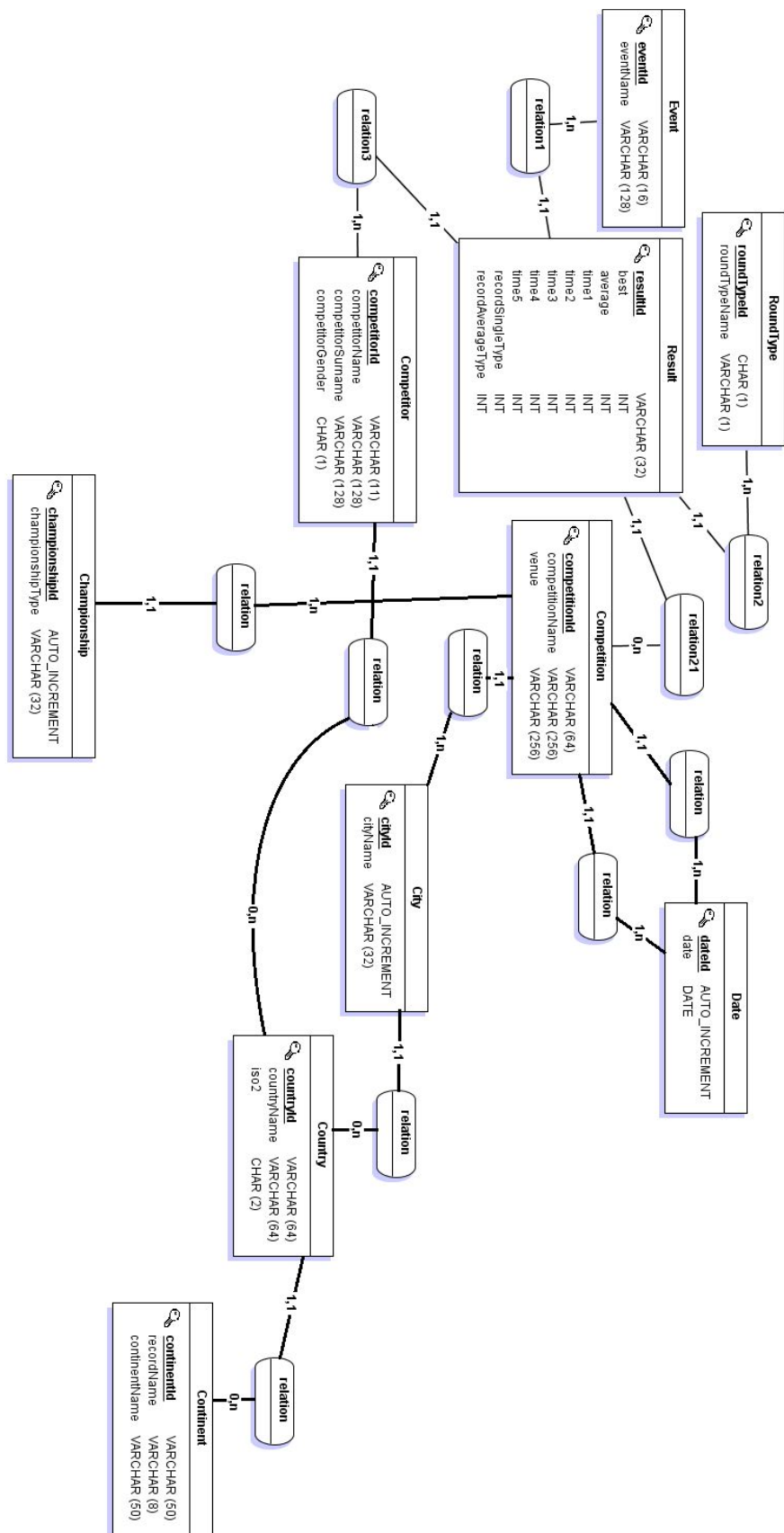
Event(eventId, eventName

PRIMARY KEY eventId)

RoundType(roundTypeId, roundTypeName

PRIMARY KEY roundTypeId)

Modèle Entité-Association :



Modèle relationnel :

Result (resultId, best, average, time1, time2, time3, time4, time5, recordSingleType, recordAverageType, *competitionId*, *eventId*, *roundTypeId*, *personId*)

Competition (competitionId, competitionName, venue, *cityId*, *dateBeginId*, *dateEndId*)

Date (dateId, date)

City (*cityId*, cityName, cityLongitude, cityLatitude, *countryId*)

Country (countryId, countryName, iso2, *continentId*)

Continent (continentId, continentName, recordName)

Championship (championshipId, championshipType, *competitionId*)

Person (personId, personName, personSurname, gender, *countryId*)

Event (eventId, eventName)

RoundType (roundTypeId, roundTypeName)

Code SQL :

```
CREATE TABLE Result(      UNSIGNED INT resultId NOT NULL AUTO_INCREMENT,
    VARCHAR(32) competitionId NOT NULL,
    VARCHAR(8) eventId NOT NULL,
    CHAR(1) roundTypeId NOT NULL,
    INT best NOT NULL,
    INT average NOT NULL,
    VARCHAR(16) personId NOT NULL,
    INT value1 NOT NULL,
    INT value2 NOT NULL,
    INT value3 NOT NULL,
    INT value4 NOT NULL,
    INT value5 NOT NULL,
    VARCHAR(32) recordSingleType,
    VARCHAR(32) recordAverageType
    PRIMARY KEY resultId,
    FOREIGN KEY competitionId REFERENCES Competition(competitionId),
    FOREIGN KEY eventId REFERENCES Event(eventId),
    FOREIGN KEY roundTypeId REFERENCES RoundType(roundTypeId),
    FOREIGN KEY personId REFERENCES Person(personId));
```

```
CREATE TABLE Competition(      VARCHAR(64) competitionId NOT NULL,
    VARCHAR(256) competitionName NOT NULL,
    UNSIGNED INT cityId NOT NULL AUTO_INCREMENT,
    DATE dateBeginId NOT NULL,
    DATE dateEndId NOT NULL,
    VARCHAR(256) venue,
    PRIMARY KEY competitionId,
    FOREIGN KEY cityId REFERENCES City(cityId),
    FOREIGN KEY dateBeginId REFERENCES Date(dateId),
    FOREIGN KEY dateEndId REFERENCES Date(dateId));
```

```
CREATE TABLE Date(      UNSIGNED INT dateId NOT NULL AUTO_INCREMENT,
    DATE date NOT NULL,
    PRIMARY KEY dateId);
```

```
CREATE TABLE City(      UNSIGNED INT cityId NOT NULL AUTO_INCREMENT,
    VARCHAR(256) cityName NOT NULL,
    VARCHAR(64) countryId NOT NULL,
    INT cityLongitude NOT NULL,
    INT cityLatitude NOT NULL,
    PRIMARY KEY cityId,
    FOREIGN KEY countryId REFERENCES Country(countryId));
```

```
CREATE TABLE Country(
    VARCHAR(64) countryId NOT NULL,
    VARCHAR(64) countryName NOT NULL,
    VARCHAR(32) continentId NOT NULL,
    VARCHAR(32) iso2 NOT NULL,
    PRIMARY KEY countryId,
    FOREIGN KEY continentId REFERENCES Continent(continentId));
```

```
CREATE TABLE Continent( VARCHAR(32) continentId NOT NULL,
    VARCHAR(4) recordName NOT NULL,
    VARCHAR(32) continentName NOT NULL,
    PRIMARY KEY continentId);
```

```
CREATE TABLE Championship(
    UNSIGNED INT championshipId NOT NULL
    AUTO_INCREMENT,
    VARCHAR(64) competitionId NOT NULL,
    VARCHAR(32) championshipType,
    PRIMARY KEY championshipId,
    FOREIGN KEY competitionId REFERENCES Competition(competitionId));
```

```
CREATE TABLE Person(    VARCHAR(10) personId NOT NULL,
    VARCHAR(64) personName NOT NULL,
    VARCHAR(64) personSurname NOT NULL,
    VARCHAR(64) countryId NOT NULL,
    CHAR(1) gender NOT NULL,
    PRIMARY KEY personId,
    FOREIGN KEY countryId REFERENCES Country(countryId));
```

```
CREATE TABLE Event(      VARCHAR(8) eventId NOT NULL,
                        VARCHAR(128) eventName NOT NULL,
                        PRIMARY KEY eventId);
```

```
CREATE TABLE RoundType(
    CHAR(1) roundTypeId NOT NULL,
    VARCHAR(32) roundTypeName NOT NULL,
    PRIMARY KEY roundTypeId);
```