# Report 1: Entity-Relationship Conceptual Modelling

The E-R components are:

**Music Band**: band-name, band webpage; each band has several releases. (1: M)

**Band Release**: title of release, year of release, release rate, release type.

**Songs:** songtitle, award; songs associate with band release, band produce songs.

**Musician**: musician name, personal website;

**Membership**: musician name, start year, end year, band name, role;

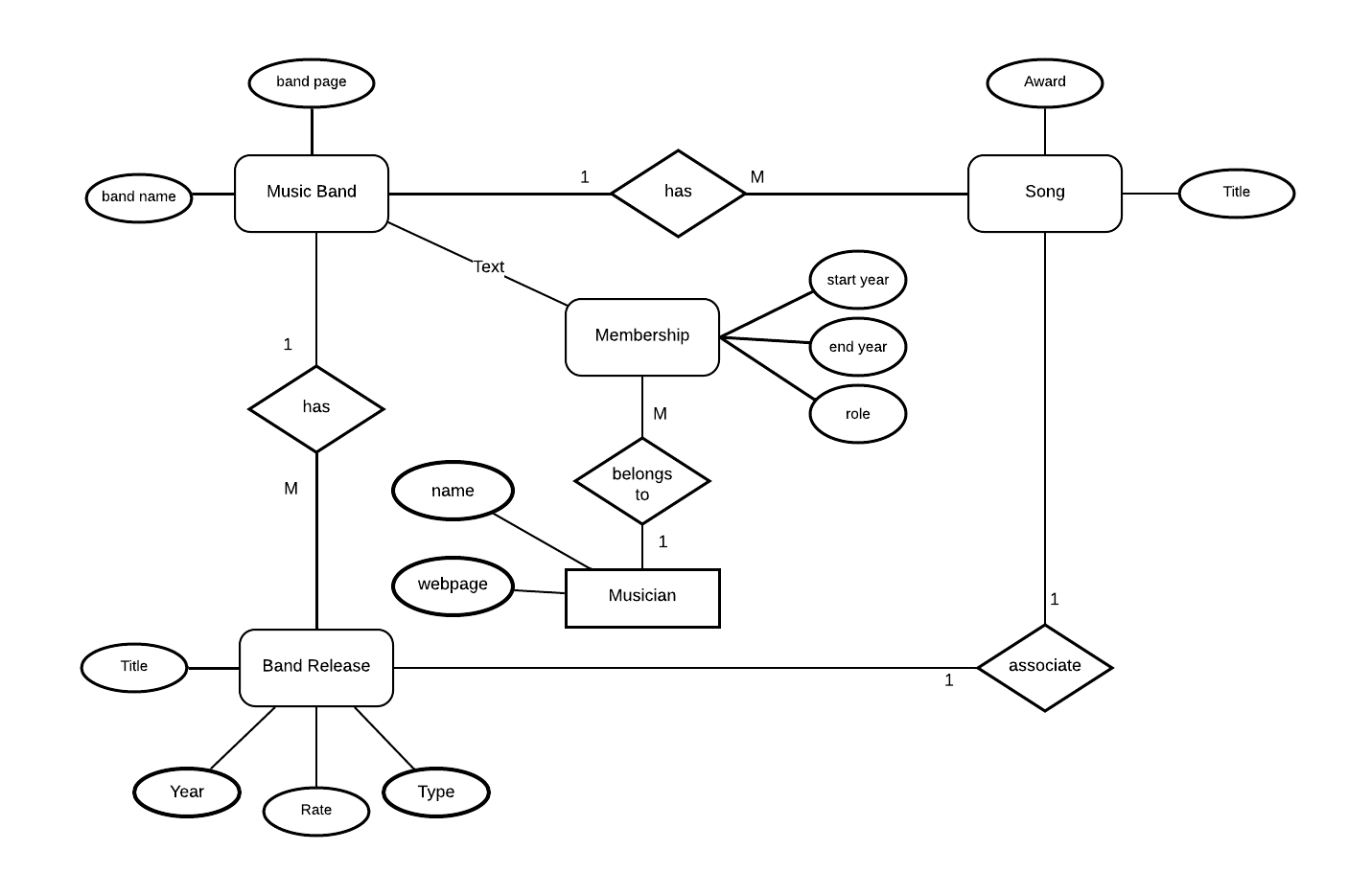


Figure 1: The E-R Diagram of “Band” Database

# Report 2: Relational Schema Modelling

## Relational Schema

**Music Band**: band-name, band webpage.

**Release**: Release-ID, release title, year of release, release rate, release type.

**Songs:** Release-ID，Songtitle, award.

**Musician**: Musician name, Personal webpage.

**Membership**: Membership-ID, Musician name, start year, end year, band name, role.

It has been claimed that the different bands may have the same band name before it was registered as a trademark, as there are millions of bands online [1]. Therefore, this report assume that the “Band” is discriminated by a combination of band name and webpage. Although a musician for a specific time period is a member of a specific band only, the time period cannot be used as a primary key, for the primary key cannot be null value. A membership ID was therefore introduced as the primary key of membership.

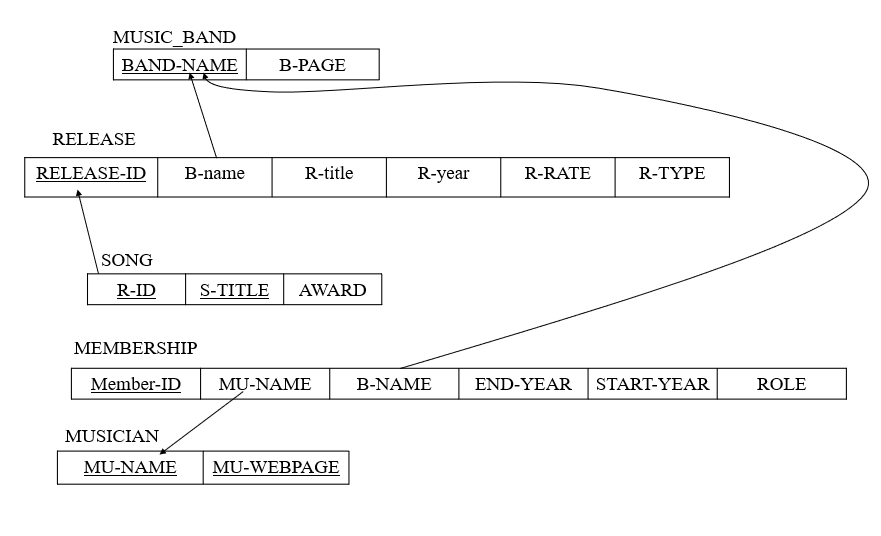


Figure 2: Relational Schema at BCNF

# Report 3.1: SQL CREATE

CREATE SCHEMA BANDS;

CREATE table MUSIC\_BAND

(Band\_name VARCHAR(32),

Band\_page VARCHAR(64),

PRIMARY KEY (Band\_name));

CREATE TABLE MUSICIAN

(Musician\_name VARCHAR(32),

Musician\_webpage VARCHAR(64),

PRIMARY KEY (Musician\_name,Musician\_webpage),

UNIQUE (Musician\_name));

CREATE TABLE RELEASE\_INFO

(Release\_ID CHAR(10),

Band\_name VARCHAR(32),

Release\_title VARCHAR(32),

Release\_year CHAR(4),

Release\_type VARCHAR(8),

Release\_rate INT,

PRIMARY KEY(Release\_ID),

FOREIGN KEY(Band\_name) REFERENCES MUSIC\_BAND(Band\_name)

ON DELETE CASCADE ON UPDATA CASCADE);

CREATE TABLE SONG

(RELEASE\_ID CHAR(10),

Song\_title CHAR(32),

Award BOOLEAN,

PRIMARY KEY (Release\_ID,Song\_title),

FOREIGN KEY(Release\_ID) REFERENCES RELEASE\_INFO(Release\_ID)

ON DELETE CASCADE ON UPDATA CASCADE);

CREATE TABLE MEMBERSHIP

(Member\_ID CHAR(10),

Musician\_name VARCHAR(32),

Band\_name VARCHAR(32),

End\_year DATE,

Start\_year DATE,

Band\_role VARCHAR(16),

PRIMARY KEY (Member\_ID),

FOREIGN KEY(Musician\_name) REFERENCES MUSICIAN(Musician\_name)

ON DELETE CASCADE ON UPDATA CASCADE,

FOREIGN KEY(Band\_name) REFERENCES MUSIC\_BAND(Band\_name));

ON DELETE CASCADE ON UPDATA CASCADE,

Explanation of constrains:

# Report 3.2: SQL SELECT

SQL 1: Show the names of all current band members of the band “the CURE”.

SELECT Musician\_name

FRROM MEMBERSHIP

WHERE Band\_name = “The Cure” AND End\_year is NULL

SQL 2: Find out if there exists a band with the same name as any member’s name of any band.

SELECT Band\_name, Musician\_name

FROM MUSIC\_BAND, MUSICIAN

WHERE Band\_name = Musician\_name

SQL 3: Show the number of members in the database who have played “bass”

SELECT COUNT (Musician\_name)

FROM MEMBERSHIP

WHERE ROLE = “bass”;

// what if two musician have a same name?

SQL 4:

SELECT Release\_name, Musician\_name, Release\_year

FROM MEMBERSHIP

WHERE Role = “bass”

ORDER BY Musician\_name, Release\_year;

<https://www.w3schools.com/sql/sql_orderby.asp>

# Report 4.1: Relational Algebra Tree

# Bibliography

|  |  |
| --- | --- |
| [1] | T. Griffin, “Mark My Word: Trademark Your Band Name,” *Texas Review of Entertainment & Sports Law,* pp. 177-192, 2014. |