Database Systems
Mini Project

Obeid, Elias
eobeid11@student.aau.dk

Caspersen, Kent
kcaspe11@student.aau.dk

Madsen, Martin
mbma11@student.aau.dk

d601f14, 1.1.01
6th February, 2014

1 Self Study 1: Preliminary Database Modeling

As stated in the assignment, we have decided to look at different possible attributes and models by looking at the structure of movie pages on IMDB. Initially, we think it would require many join tables, as we've identified a few many-to-many relationships among structures we've discussed. These structures are: *actors*, *directors*, *writers*, *movies*, *awards*, *ratings*, and *users*.

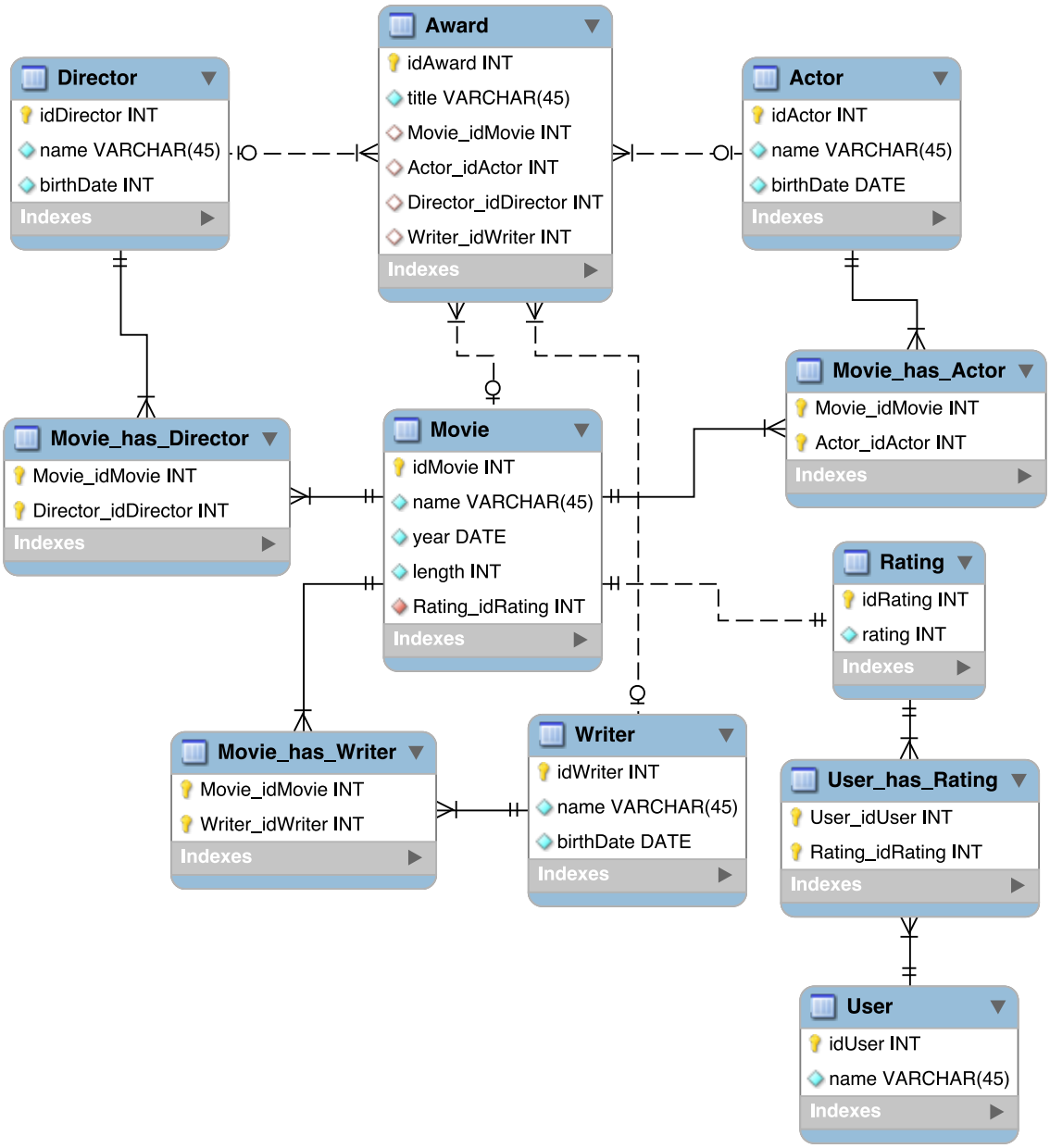

Figure 1: *Enhanced entity-relationship (EER) model diagram of a simplified movie database.*

We spent time on figuring out how to map the relationships between tables instead of focusing on the attributes. In our opinion it is easy to just add a birthdate if that should be necessary.

Figure 1 shows relationships between the chosen models and their corresponding join tables. Dashed lines between tables represent *non-identifying* relationships and solid lines between tables represent *identifying* relationships.

When lines branch toward a table then there is a "has many" relationship to that table. When the lines have two orthogonal dashes (or a orthogonal dash and a circle) by a table then there is a "has one" relationship to that table. If there is a circle then the relationship is non-identifying. For example one *Director* has many *Awards*. The relationship is also non-identifying because the tables can exist indenpendently of each other.

The SQL for creating this structure is presented in see section A.

A SQL: Preliminary Database Modeling

```
SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';

CREATE SCHEMA IF NOT EXISTS `mydb` DEFAULT CHARACTER SET utf8 COLLATE
  utf8_general_ci ;
USE `mydb` ;

-- -----------------------------------------------------
-- Table `mydb`.`Rating`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Rating` (
  `idRating` INT NOT NULL,
  `rating` INT NOT NULL,
  PRIMARY KEY (`idRating`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Movie`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Movie` (
  `idMovie` INT NOT NULL,
  `name` VARCHAR(45) NOT NULL,
  `year` DATE NOT NULL,
  `length` INT NOT NULL,
  `Rating_idRating` INT NOT NULL,
  PRIMARY KEY (`idMovie`),
  UNIQUE INDEX `name_UNIQUE` (`name` ASC),
  INDEX `fk_Movie_Rating1_idx` (`Rating_idRating` ASC),
  CONSTRAINT `fk_Movie_Rating1`
    FOREIGN KEY (`Rating_idRating`)
    REFERENCES `mydb`.`Rating` (`idRating`)
```

```
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Actor`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Actor` (
  `idActor` INT NOT NULL,
  `name` VARCHAR(45) NOT NULL,
  `birthDate` DATE NOT NULL,
  PRIMARY KEY (`idActor`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Director`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Director` (
  `idDirector` INT NOT NULL,
  `name` VARCHAR(45) NOT NULL,
  `birthDate` INT NOT NULL,
  PRIMARY KEY (`idDirector`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Writer`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Writer` (
  `idWriter` INT NOT NULL,
  `name` VARCHAR(45) NOT NULL,
  `birthDate` DATE NOT NULL,
  PRIMARY KEY (`idWriter`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Award`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Award` (
  `idAward` INT NOT NULL,
  `title` VARCHAR(45) NOT NULL,
  `Movie_idMovie` INT NULL,
  `Actor_idActor` INT NULL,
  `Director_idDirector` INT NULL,
  `Writer_idWriter` INT NULL,
```

```
  PRIMARY KEY (`idAward`),
  INDEX `fk_Award_Movie1_idx` (`Movie_idMovie` ASC),
  INDEX `fk_Award_Actor1_idx` (`Actor_idActor` ASC),
  INDEX `fk_Award_Director1_idx` (`Director_idDirector` ASC),
  INDEX `fk_Award_Writer1_idx` (`Writer_idWriter` ASC),
  CONSTRAINT `fk_Award_Movie1`
    FOREIGN KEY (`Movie_idMovie`)
    REFERENCES `mydb`.`Movie` (`idMovie`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Award_Actor1`
    FOREIGN KEY (`Actor_idActor`)
    REFERENCES `mydb`.`Actor` (`idActor`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Award_Director1`
    FOREIGN KEY (`Director_idDirector`)
    REFERENCES `mydb`.`Director` (`idDirector`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Award_Writer1`
    FOREIGN KEY (`Writer_idWriter`)
    REFERENCES `mydb`.`Writer` (`idWriter`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`User`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`User` (
  `idUser` INT NOT NULL,
  `name` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`idUser`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Movie_has_Writer`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Movie_has_Writer` (
  `Movie_idMovie` INT NOT NULL,
  `Writer_idWriter` INT NOT NULL,
  PRIMARY KEY (`Movie_idMovie`, `Writer_idWriter`),
  INDEX `fk_Movie_has_Writer_Writer1_idx` (`Writer_idWriter` ASC),
  INDEX `fk_Movie_has_Writer_Movie1_idx` (`Movie_idMovie` ASC),
  CONSTRAINT `fk_Movie_has_Writer_Movie1`
```

```
    FOREIGN KEY (`Movie_idMovie`)
    REFERENCES `mydb`.`Movie` (`idMovie`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Movie_has_Writer_Writer1`
    FOREIGN KEY (`Writer_idWriter`)
    REFERENCES `mydb`.`Writer` (`idWriter`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Movie_has_Actor`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Movie_has_Actor` (
  `Movie_idMovie` INT NOT NULL,
  `Actor_idActor` INT NOT NULL,
  PRIMARY KEY (`Movie_idMovie`, `Actor_idActor`),
  INDEX `fk_Movie_has_Actor_Actor1_idx` (`Actor_idActor` ASC),
  INDEX `fk_Movie_has_Actor_Movie1_idx` (`Movie_idMovie` ASC),
  CONSTRAINT `fk_Movie_has_Actor_Movie1`
    FOREIGN KEY (`Movie_idMovie`)
    REFERENCES `mydb`.`Movie` (`idMovie`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Movie_has_Actor_Actor1`
    FOREIGN KEY (`Actor_idActor`)
    REFERENCES `mydb`.`Actor` (`idActor`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`Movie_has_Director`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Movie_has_Director` (
  `Movie_idMovie` INT NOT NULL,
  `Director_idDirector` INT NOT NULL,
  PRIMARY KEY (`Movie_idMovie`, `Director_idDirector`),
  INDEX `fk_Movie_has_Director_Director1_idx` (`Director_idDirector` ASC),
  INDEX `fk_Movie_has_Director_Movie1_idx` (`Movie_idMovie` ASC),
  CONSTRAINT `fk_Movie_has_Director_Movie1`
    FOREIGN KEY (`Movie_idMovie`)
    REFERENCES `mydb`.`Movie` (`idMovie`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
```

```
  CONSTRAINT `fk_Movie_has_Director_Director1`
    FOREIGN KEY (`Director_idDirector`)
    REFERENCES `mydb`.`Director` (`idDirector`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `mydb`.`User_has_Rating`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`User_has_Rating` (
  `User_idUser` INT NOT NULL,
  `Rating_idRating` INT NOT NULL,
  PRIMARY KEY (`User_idUser`, `Rating_idRating`),
  INDEX `fk_User_has_Rating_Rating1_idx` (`Rating_idRating` ASC),
  INDEX `fk_User_has_Rating_User1_idx` (`User_idUser` ASC),
  CONSTRAINT `fk_User_has_Rating_User1`
    FOREIGN KEY (`User_idUser`)
    REFERENCES `mydb`.`User` (`idUser`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_User_has_Rating_Rating1`
    FOREIGN KEY (`Rating_idRating`)
    REFERENCES `mydb`.`Rating` (`idRating`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
ENGINE = InnoDB;


SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```