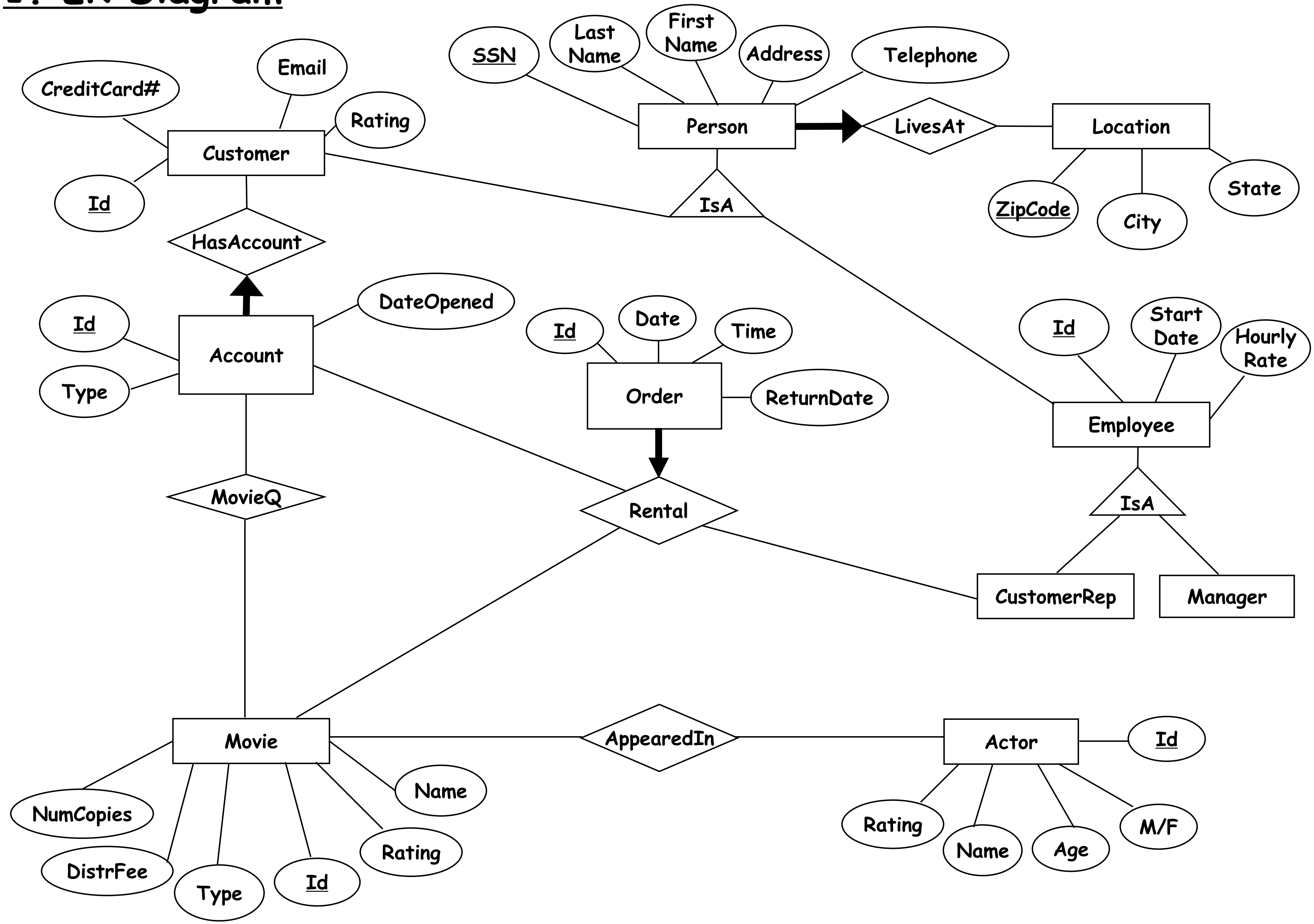
CSE 305
Spring 2017
Course Project Assignment #1
(ER & Relational Models for Online Movie Rental System)
Solution

Feb. 18, 2017

1. ER Diagram



SSN
Last Name
First Name
Address
Telephone
Email
CreditCard#
Rating
Customer
Person
LivesAt
Location
Id
IsA
ZipCode
City
State
HasAccount
DateOpened
Id
Account
Type
Id
Date
Time
Order
ReturnDate
Id
Start Date
Hourly Rate
Employee
MovieQ
Rental
IsA
CustomerRep
Manager
Movie
AppearedIn
Actor
Id
NumCopies
Name
Rating
DistrFee
Type
Id
Rating
Name
Age
M/F

Rationale for E-R Model

Rental information is primarily specified using three entity sets: **Account**, **Movie**, and **Order**. Entities of these types are linked through the relationship set **MovieQ**, which represents the queue of movies the account owner is interested in renting, and the relationship set **Rental**, which represents the actual renting of movies. Note that due to the single-role key constraint and participation constraint that the **Order** entity set imposes on the **Rental** relationship set, there is a one-to-one correspondence between **Order** entities and **Rental** relationships.

A customer's rental history is not represented explicitly in this model. Rather, it is encapsulated in the **Order** entity set: a return date of null indicates that the movie in question has not yet been returned.

Note that a customer can open multiple accounts, each of which can be of a different type (limited, unlimited, etc.).

2. Relational Model (1)

```
CREATE TABLE Person (
    SSN          INTEGER,
    LastName     CHAR(20) NOT NULL,
    FirstName    CHAR(20) NOT NULL,
    Address      CHAR(20),
    ZipCode      INTEGER,
    Telephone    INTEGER,
    PRIMARY KEY (SSN),
    FOREIGN KEY (ZipCode) REFERENCES Location (ZipCode)
            ON DELETE NO ACTION
            ON UPDATE CASCADE     )


CREATE TABLE Location (
    ZipCode  INTEGER,
    City     CHAR(20) NOT NULL,
    State    CHAR(20) NOT NULL,
             PRIMARY KEY (ZipCode)  )


CREATE TABLE Employee (
    ID            EmpId,
    SSN           INTEGER,
    StartDate     DATE,
    HourlyRate    INTEGER,
    PRIMARY KEY (ID),
    FOREIGN KEY (SSN) REFERENCES Person (SSN))
            ON DELETE NO ACTION
            ON UPDATE CASCADE     )
```

2. Relational Model (2)

```
CREATE TABLE Account (
            Id              INTEGER,
            DateOpened DATE,
            Type            AccountType,
            Customer     CustomerId,
            PRIMARY KEY (Id),
            FOREIGN KEY (Customer) REFERENCES Customer (Id)
                            ON DELETE NO ACTION
                            ON UPDATE CASCADE        )

CREATE TABLE Customer (
            Id              CustomerId,
            Email           CHAR(32),
            Rating          INTEGER,
            CreditCardNumber INTEGER,
            PRIMARY KEY (Id),
            FOREIGN KEY (Id) REFERENCES Person (SSN)
                            ON DELETE NO ACTION
                            ON UPDATE CASCADE        )

CREATE TABLE Order (
            Id              INTEGER,
            DateTime     DATETIME,
            ReturnDate  DATE,
            PRIMARY KEY (Id)      )
```

2. Relational Model (3)

```
CREATE TABLE Movie (
            Id              INTEGER,
            Name            CHAR(20) NOT NULL,
            Type            CHAR(20) NOT NULL,
            Rating          INTEGER,
            DistrFee        CURRENCY,
            NumCopies       INTEGER,
            PRIMARY KEY (Id)        )

CREATE TABLE Actor (
            Id              INTEGER,
            Name            CHAR(20) NOT NULL,
            Age             INTEGER NOT NULL,
            M/F             CHAR(1) NOT NULL,
            Rating          INTEGER,
            PRIMARY KEY (Id)        )


CREATE DOMAIN AccountType CHAR(20)
            CHECK (  VALUE IN ('Limited', 'Unlimited-1', 'Unlimited-2', 'Unlimited-3')  )

CREATE DOMAIN EmpId INTEGER
   CHECK (EmpId  > 0 AND EmpId  < 1000000000)

CREATE DOMAIN CustomerId INTEGER
            CHECK (CustomerId  > 0 AND CustomerId  < 1000000000)
```

2. Relational Model (4)

```
CREATE TABLE Rental (
        AccountId   INTEGER,
        CustRepId   EmpId,
        OrderId     INTEGER,
        MovieId     CHAR(20),
        PRIMARY KEY (AccountId, CustRepId, OrderId, MovieId),
        FOREIGN KEY (AccountId) REFERENCES Account (Id)
                    ON DELETE NO ACTION
                    ON UPDATE CASCADE,
        FOREIGN KEY (CustRepId) REFERENCES Employee (Id)
                    ON DELETE NO ACTION
                    ON UPDATE CASCADE,
        FOREIGN KEY (OrderId) REFERENCES Order (Id)
                    ON DELETE NO ACTION
                    ON UPDATE CASCADE
        FOREIGN KEY (MovieId) REFERENCES Movie (Id)
                    ON DELETE NO ACTION
                    ON UPDATE CASCADE        )
```

2. Relational Model (5)

```
CREATE TABLE MovieQ (
        AccountId   INTEGER,
        MovieId     CHAR(20),
        PRIMARY KEY (AccountId, MovieId),
        FOREIGN KEY (AccountId) REFERENCES Account (Id)
                    ON DELETE NO ACTION
                    ON UPDATE CASCADE,
                    FOREIGN KEY (MovieId) REFERENCES Movie (Id)
                    ON DELETE NO ACTION
                    ON UPDATE CASCADE       )

CREATE TABLE AppearedIn (
        ActorId     INTEGER,
        MovieId     CHAR(20),
        PRIMARY KEY (ActorId, MovieId),
        FOREIGN KEY (ActorId) REFERENCES Actor (Id)
                    ON DELETE NO ACTION
                    ON UPDATE CASCADE,
                    FOREIGN KEY (MovieId) REFERENCES Movie (Id)
                    ON DELETE NO ACTION
                    ON UPDATE CASCADE       )
```