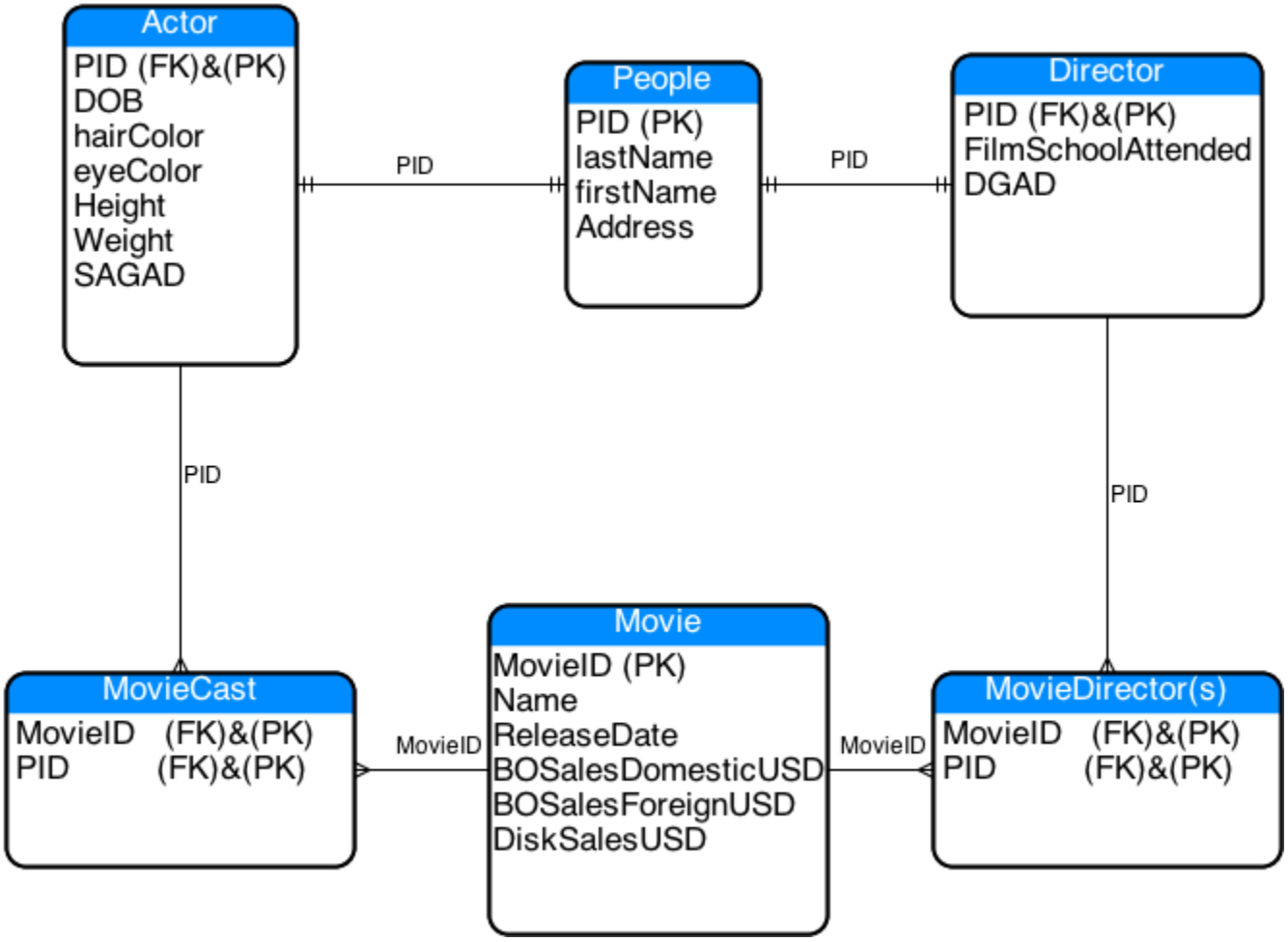
To EON Productions,

Here is the database design.

E-R diagram:




```
--SQL statements for EON productions database
--Author: Anthony Cali

--Create the people table
drop table if exists people;
create table people (
   pid             serial   not null   primary key,
   lastName        text      not null,
   firstName       text      not null,
   address         text      not null
 );

-- FD: PID -> Name(firstname, lastname), address
```

```
--Create the actor table
drop table if exists actor;
create table actor (
   pid          int      not null references people(pid),
   DOB          date     not null,
   hairColor text     not null,
   eyeColor   text     not null,
   height_in int      not null,
   weight_lb int      not null,
   SAGAD       date,
   primary key (pid)
);

--FD: PID -> DOB, hairColor, eyeColor, height, weight, SAGAD

--Create the director table
drop table if exists director;
create table director (
   pid                    int      not null references people(pid),
   F_SchoolAttended   text     not null,
   DGAD                   date,
   primary key (pid)
);

--FD: PID -> F_SchoolAttended, DGAD

--Create the movies table
drop table if exists movies;
create table movies (
   movieID                serial not null primary key,
   name                   text     not null,
   releaseDate            date     not null,
   BOSalesDomesticUSD int      not null,
   BOSalesForiegnUSD   int      not null,
   diskSalesUSD          int      not null
);

--FD: movieID -> name, releaseDate, BOSalesDomesticUSD, BOSalesForeignUSD,
--                   diskSalesUSD

--Create the movieDirectors table
drop table if exists movieDirectors;
create table movieDirectors (
   movieID int not null references movies(movieID),
   pid       int not null references director(pid),
```

```
   primary key (movieID, pid)
);

--FD: (movieID, pid) ->

--create the cast tabe
drop table if exists movieCast;
create table movieCast (
   movieID int not null references movies(movieID),
   pid       int not null references actor(pid),
   primary key (movieID, pid)
);

--FD: (movieID, pid) ->

--Add test data

insert into people (firstName, lastName, address)
values ('Sean', 'Connery', 'Bahamas'),
('Woody', 'Allen', 'New York City'),
('Pierce', 'Brosnan', 'Cape Town'),
('John', 'Boorman', 'Ireland'),
('Martin', 'Campbell','London'),
('Terence', 'Young', 'Alpes-Maritimes');

insert into movies (name, releaseDate, BOSalesDomesticUSD, BOSalesForiegnUSD,
                    diskSalesUSD)
values ('Dr. No', '1963-05-08', 16067035, 43500000, 23840000),
('Annie Hall', '1977-04-20', 38251425, 797689, 19002366),
('Zardoz', '1974-02-06', 0,0,1800000),
('Golden Eye', '1995-10-17', 107294034,244900000, 46099300);

insert into actor (pid, DOB, hairColor, eyeColor, height_in, weight_lb, SAGAD)
values (1, '1930-08-25', 'Black', 'Brown', 74, 180, '1987-01-13'),
(2,'1935-12-01', 'Brown', 'Brown', 65 , 150, '1978-01-13'),
(3,'1953-05-16', 'Black', 'Blue', 73, 164, '1985-01-13');

insert into director(pid, F_SchoolAttended, DGAD)
values (6, 'Oxford', '1949-01-01'),
(2, 'New York University', '1978-01-13'),
(5, 'Not Applicable', '1998-01-13'),
(4, 'Salesian School', '1973-01-13');

insert into movieDirectors (movieID, pid)
values (1,6),
(2,2),
```

```
(3,4),
(4,5);

insert into movieCast (movieID, pid)
values (1,1),
(2,2),
(3,1),
(4,3);

--end of test data

--Query on the director(s) who have worked with Sean Connery as per
--   the dataset represented.

select p2.firstName as firstName, p2.lastName as lastName
   from movieDirectors md inner join people p2
      on md.pid = p2.pid
 where md.movieID in
(     select mc.movieID
         from movieCast mc inner join people p
            on p.pid = mc.pid
         where p.firstName = 'Sean'
           and p.lastName = 'Connery'
);
```