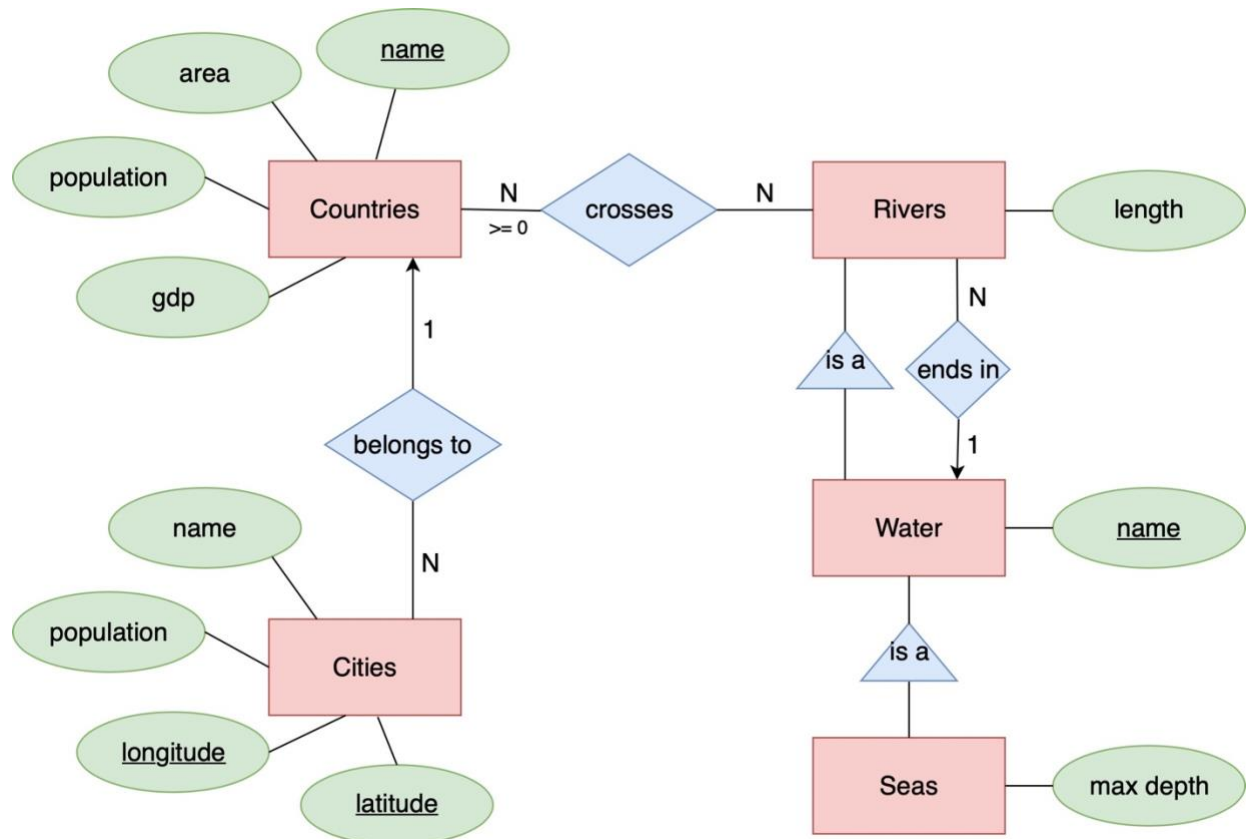


DATA 514 Homework 4: Entity Relationship Diagrams and Query Plans  
Cathy Jia

Part 1: Theory

1. Design an E/R diagram for geography that contains the objects or entities together with the listed attributes.



2. Translate the diagram above by writing the SQL CREATE TABLE statements to represent this E/R diagram.

```
CREATE TABLE Person (  
  ssn INT,  
  name VARCHAR(20),  
  PRIMARY KEY (ssn)  
);
```

```
CREATE TABLE Driver (  
  driverID INT,  
  driverSSN INT,
```

```
    PRIMARY KEY (driverID),  
    FOREIGN KEY (driverSSN) REFERENCES Person  
);
```

```
CREATE TABLE NonProfessionalDriver(  
    driverID INT,  
    PRIMARY KEY (driverID),  
    FOREIGN KEY (driverID) REFERENCES Driver  
);
```

```
CREATE TABLE ProfessionalDriver(  
    driverID INT,  
    medicalHistory VARCHAR(20),  
    PRIMARY KEY (driverID),  
    FOREIGN KEY (driverID) REFERENCES Driver  
);
```

```
CREATE TABLE InsuranceCo (  
    name VARCHAR(20),  
    phone INT,  
    PRIMARY KEY (name)  
);
```

```
CREATE TABLE Vehicle (  
    licensePlate VARCHAR(20),  
    year INT,  
    insuranceName VARCHAR(20),  
    maxLiability FLOAT,  
    ownerSSN INT,  
    PRIMARY KEY (licensePlate),  
    FOREIGN KEY (insuranceName) REFERENCES InsuranceCo,  
    FOREIGN KEY (ownerSSN) REFERENCES Person  
);
```

```
CREATE TABLE Car (  
    carLicensePlate VARCHAR(20),  
    make VARCHAR(20),  
    PRIMARY KEY (carLicensePlate),  
    FOREIGN KEY (carLicensePlate) REFERENCES Vehicle  
);
```

```
CREATE TABLE CarDriver (  
    carLicensePlate VARCHAR(20),  
    carDriverID INT,
```

```

PRIMARY KEY (carLicensePlate),
FOREIGN KEY (carLicensePlate) REFERENCES Vehicle,
FOREIGN KEY (carDriverID) REFERENCES NonProfessionalDriver
);

```

```

CREATE TABLE Truck (
truckLicensePlate VARCHAR(20),
truckDriverID INT,
capacity VARCHAR(20),
PRIMARY KEY (truckLicensePlate),
FOREIGN KEY (truckLicensePlate) REFERENCES Vehicle,
FOREIGN KEY (truckDriverID) REFERENCES ProfessionalDriver
);

```

**Which relation in your relational schema represents the relationship "insures" in the E/R diagram and why is that your representation?**

The relationship "insures" is represented by the name of insurance company and the maximum liability for the vehicle entity. The vehicle entity has a column insuranceName which is a foreign key referencing the name in the insuranceCo entity, and the column maxLiability. I make this representation because each vehicle can only hold one insurance information including the insurance company name and maximum liability.

**Compare the representation of the relationships "drives" and "operates" in your schema, and explain why they are different.**

In the relationship "drives", a car can have multiple drivers, so I represent the Car entity (carLicensePlate, make) and the CarDriver entity (carLicensePlate, carDriverID) separately. In the relationship "operates", a truck can be operated by only one driver, so I represent the Truck entity including both truck information and truck driver information (truckLicensePlate, truckDriverID, capacity).

**3. Write the equivalent SQL query to the relational algebra query plan.**

```

SELECT d.p1 AS p1, MAX(d.y) AS z
FROM (SELECT a.p1 AS p1, b.p3 AS p3, COUNT(*) AS y
      FROM (SELECT *
            FROM person_living AS pl, parent_child AS pc
            WHERE pl.x = pc.p1) AS a,
            (SELECT p1 AS p3, p2 AS p4
            FROM parent_child) AS b
      WHERE a.p2 = b.p3
      GROUP BY a.p1, b.p3) AS d
GROUP BY d.p1

```

#### 4. Write a relational algebra plan

$T1(p) = \text{person\_living}(p) \text{ Join}[p=\text{name}] \text{ male}(\text{name})$

$T2(p1,p2) = \text{parent\_child}(p1,p2) \text{ Join}[p1=\text{name}] \text{ female}(\text{name})$

$T3(p) = T2(p1,p2) \text{ Join}[p2=p] T1(p)$

$T4(p) = T1(p) \text{ Difference } T3(p)$