Technologies for Information Systems

Part I (10 points)

prof. L. Tanca – September 21st, 2016

Available time: 25 minutes

Last Name ____

First Name ____

Student ID ________________ **Signature**____

1) Define Wrappers and Mediators, explain in which circumstances their use is advised in Data Integration and the way they work. Discuss the various types of Wrappers and Mediators that have been introduced during the course.

2) Describe the Box-Plot method for displaying the distribution of data, using an example to illustrate it clearly.

Technologies for Information Systems

Part II (22 points)

prof. L. Tanca – September 21st, 2016

Available time: 2h 00m

Last Name ____

First Name ____

Student ID ________________ **Signature**____

PoliRestaurants is a restaurant chain selling home-delivery meals in Italy through its website. Each order is associated with just one dish, and the customers may choose the restaurant to which the delivery is requested. Each dish has a full price, but a discount may be applied when the customer completes the order; the restaurant chain proposes promotions varying continuously, therefore each order is associated with its own discount.
The management of *PoliRestaurants* has now hired you to design a data warehouse to analyze the orders.

The following is the schema of the *PoliRestaurants* operational database:

RESTAURANT (<u>RestaurantName</u>, Address, CityName, SupervisorSSN, NumberOfEmployees)
SUPERVISOR (<u>SupervisorSSN</u>, Surname, GivenName)
CITY (<u>CityName</u>, Region)
DISHSUBCATEGORY (<u>Subcategory</u>, Category) *// The available categories are 'Food' and 'Beverage'. Possible subcategories are 'AlcoholicDrink', 'Meat', 'Fish', 'Pasta', …*
DISH (<u>DishName</u>, Description, Subcategory, Price)
CUSTOMER (<u>CustomerId</u>, Surname, GivenName, BirthYear)
ORDER (<u>CustomerId</u>, <u>Date</u>, <u>Time</u>, DishName, RestaurantName, Discount, DeliveryAddress, DeliveryCity) *// The discount is a percentage expressed as a number between 0 and 1.*

1. (3 points) Perform the reverse engineering of the given logical schema into a conceptual schema (Entity-Relationship model).

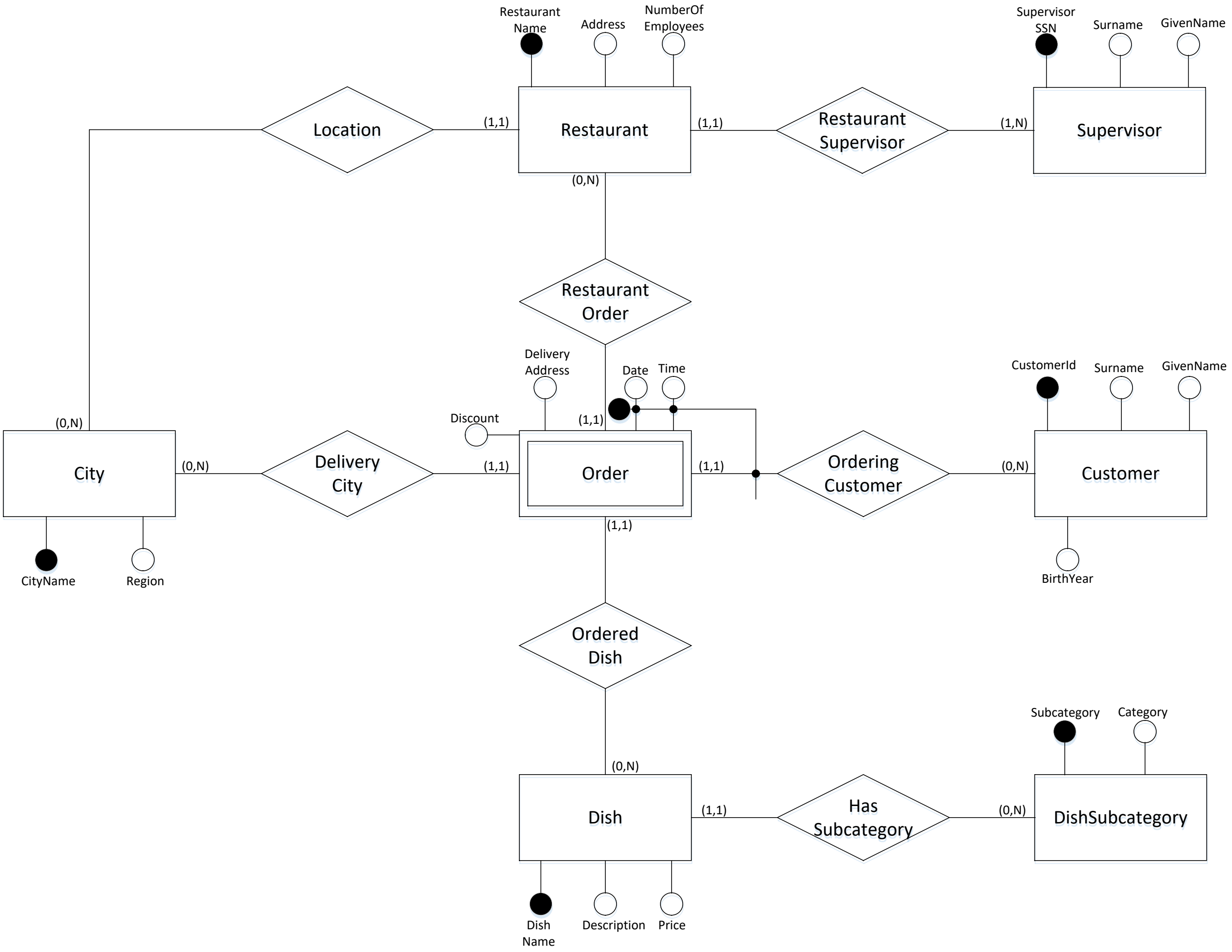
2. With respect to the produced ER diagram, discover the fact(s) that are useful specifically for answering the queries reported below. For each of these facts:
   a. (3 points) Produce the attribute tree (with pruning and grafting).
   b. (3 points) Produce the conceptual schema (fact schema).
   c. (2 points) Identify the measure(s) and produce the glossary.

3. (3 points) Produce a logical schema consistent with the conceptual schema.

4. Write in SQL the following queries against the designed logical schema:
   a. (2 points) Considering only the orders with category ‘Food’, find the average discount by customer birth year, dish subcategory and restaurant city.
   b. (2 points) Considering only the restaurants located in Lombardy, find the total number of orders by restaurant, delivery city and date. Include in the answer also the aggregations computed using only one or two of the three attributes.
   c. (2 points) Find SSN, surname and given name of the supervisors who in 2015 have increased their income at least of 30% with respect to 2014.
   d. (2 points) For each delivery region, find the dish(es) with the greatest number of orders.

SOLUTION

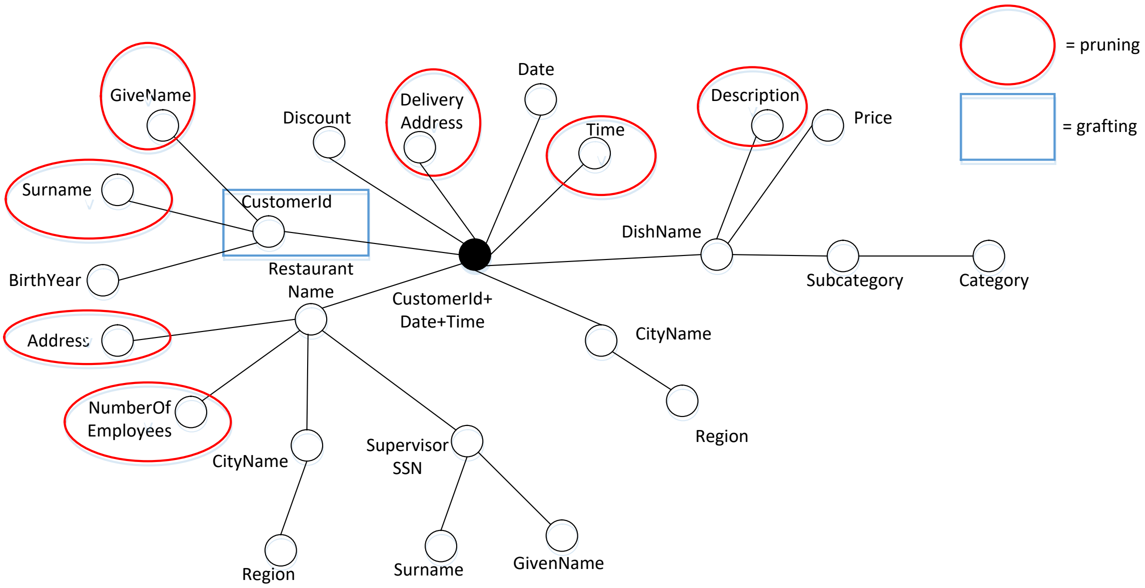
1. Reverse engineering



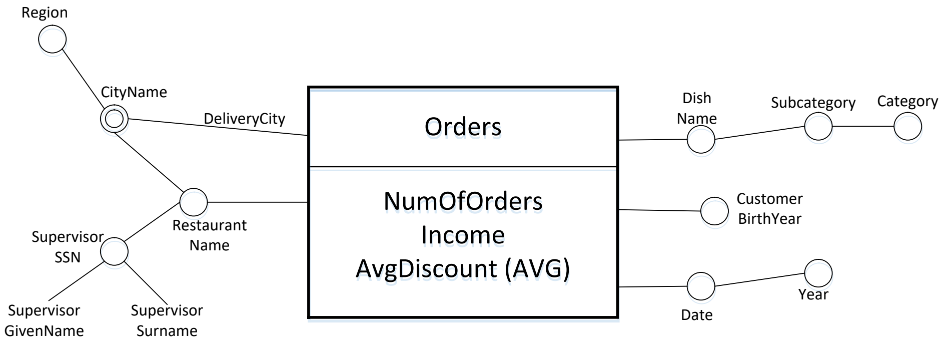
2. Conceptual design

Fact: Orders (Order entity)

2a) Attribute tree



2b) Fact schema



2c) Glossary

<u>NumOfOrders</u>

SELECT O.Date, C.BirthYear, O.DishName, O.RestaurantName, O.DeliveryCity, **COUNT**(*)
FROM Order **AS** O, Customer **AS** C
WHERE O.CustomerId=C.CustomerId
GROUP BY O.Date, C.BirthYear, O.DishName, O.RestaurantName, O.DeliveryCity

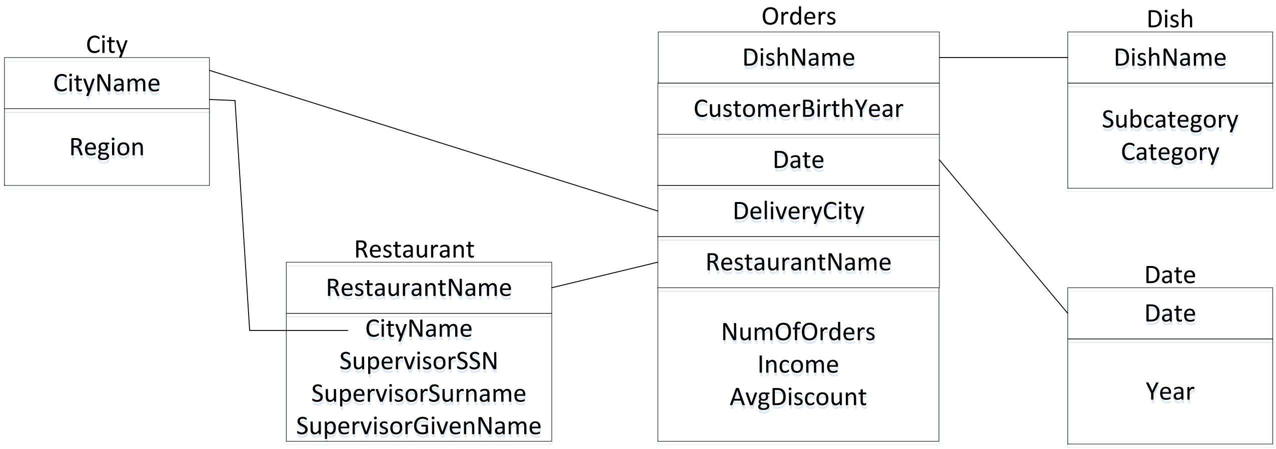
<u>Income</u>

SELECT O.Date, C.BirthYear, O.DishName, O.RestaurantName, O.DeliveryCity,
  SUM(D.Price-D.Price*O.Discount)
FROM Order **AS** O, Customer **AS** C, Dish **AS** D
WHERE O.CustomerId=C.CustomerId **AND** O.DishName=D.DishName
GROUP BY O.Date, C.BirthYear, O.DishName, O.RestaurantName, O.DeliveryCity

<u>AvgDiscount</u>

SELECT O.Date, C.BirthYear, O.DishName, O.RestaurantName, O.DeliveryCity, **AVG**(O.Discount)
FROM Order **AS** O, Customer **AS** C
WHERE O.CustomerId=C.CustomerId
GROUP BY O.Date, C.BirthYear, O.DishName, O.RestaurantName, O.DeliveryCity

3. Logical design



4. Query answering

4a) Considering only the orders with category 'Food', find the average discount by customer birth year, dish subcategory and restaurant city.

SELECT O.CustomerBirthYear, D.Subcategory, R.CityName,
  SUM(O.NumOfOrders*O.AvgDiscount)/**SUM**(O.NumOfOrders)
FROM Orders **AS** O, Dish **AS** D, Restaurant **AS** R
WHERE O.DishName=D.DishName **AND** O.RestaurantName=R.RestaurantName **AND** D.Category='Food'
GROUP BY O.CustomerBirthYear, D.Subcategory, R.CityName

4b) Considering only the restaurants located in Lombardy, find the total number of orders by restaurant, delivery city and date. Include in the answer also the aggregations computed using only one or two of the three attributes.

SELECT R.RestaurantName, O.DeliveryCity, O.Date, **SUM**(O.NumOfOrders)
FROM Orders **AS** O, Restaurant **AS** R, City **AS** C
WHERE O.RestaurantName=R.RestaurantName **AND** R.CityName=C.CityName **AND** C.Region='Lombardy'
GROUP BY R.RestaurantName, O.DeliveryCity, O.Date **WITH CUBE**

4c) Find SSN, surname and given name of the supervisors who in 2015 have increased their income at least of 30% with respect to 2014.

```
SELECT R.SupervisorSSN, R.SupervisorSurname, R.SupervisorGivenName
FROM Orders AS O, Restaurant AS R, Date AS D
WHERE O.RestaurantName=R.RestaurantName AND O.Date=D.Date AND D.Year='2015'
GROUP BY R.SupervisorSSN, R.SupervisorSurname, R.SupervisorGivenName
HAVING SUM(O.Income)>=1.3*(
                SELECT SUM(O.Income)
                FROM Orders AS O2, Restaurant AS R2, Date AS D2
                WHERE O2.RestaurantName=R2.RestaurantName AND O2.Date=D2.Date AND
                        D2.Year='2014' AND R2.SupervisorSSN=R.SupervisorSSN
                )
```

4d) For each delivery region, find the dish(es) with the greatest number of orders.

```
CREATE VIEW DeliveryRegionDishOrders (DeliveryRegion, DishName, NumOfOrders) (
        SELECT C.Region, O.DishName, SUM(O.NumOfOrders)
        FROM Orders AS O, City AS C
        WHERE O.DeliveryCity=C.CityName
        GROUP BY C.Region, O.DishName
)

SELECT D.DeliveryRegion, D.DishName
FROM DeliveryRegionDishOrders AS D
WHERE D.NumOfOrders = (
                        SELECT MAX(D2.NumOfOrders)
                        FROM DeliveryRegionDishOrders AS D2
                        WHERE D2.DeliveryRegion=D.DeliveryRegion
                        )
```