Nate Warner CS 466 Fall 2024

Pset 4 - Due: Wednesday, October 2

1. Convert the following ER diagram, which models the operational data for a department store, to a set of relations that conform to Third Normal Form (3NF)

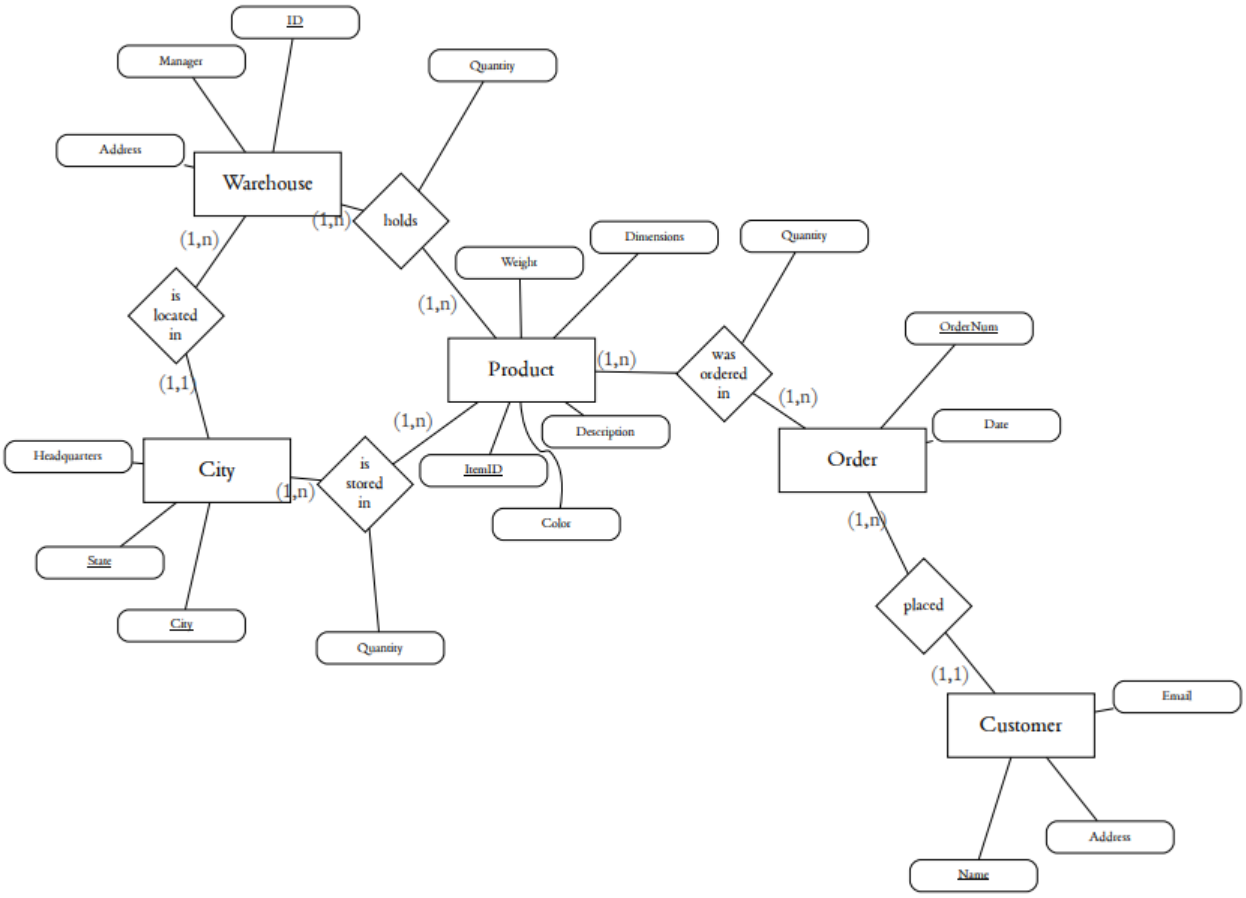

Figure 1: *Entity relationship diagram*

1.) First, we convert the strong, non-subtype entities to relations. We have

Warehouse(ID, Address, Manager)
City(State, City, Headquarters)
Product(ItemID, Color, Description, Weight, Dimensions)
Order(OrderNum, Date)
Customer(Name, Address, Email).

Since there are no subtype or weak entities, we move to the relationships. There are 5 relationships, two binary one-to-many, and 3 binary many-to-many. We start with the simpler.

To handle the binary one-to-many relationships, we insert a foreign key in the many entity to the one entity. Thus,

Warehouse(ID, Address, Manager, City†)
City(State, City, Headquarters)
Order(OrderNum, Date, Name†)
Customer(Name, Address, Email).

Where city in the warehouse relation is a foreign key to the city relation, and name in the order relation is a foreign key to the customer relation. The implied functional dependencies are

$$\text{ID} \rightarrow \text{City}$$
$$\text{OrderNum} \rightarrow \text{Name}.$$

Next, we need to handle the remaining relationships, which are the three binary many-to-many. For these, we create a new relation. The primary key of the new relations will be the concatenation of the primary keys of the entity relations, and the intersection data as non-prime attributes

Warehouse(ID, Address, Manager, City†)
City(State, City, Headquarters)
Product(ItemID, Color, Description, Weight, Dimensions)
Order(OrderNum, Date, Name†)
Customer(Name, Address, Email)
ProductHolding(ID†, ItemID†, quantity)
ProductStoredIn(City†, ItemID†, Quantity)
ProductOrderedIn(ItemID†, OrderNum†, Quantity).

The ProductHolding relation has a foreign key *ID* to the Warehouse relation, and foreign key *ItemID* to the Product relation.

The ProductStoredIn relation has a foreign key *City* to the City relation, and a foreign key *ItemID* to the Product relation.

The ProductOrderedIn relation has a foreign key *ItemID* to the Product relation, and a foreign key *OrderNum* too the Order relation.

Thus, the final schema of the database is the following eight relations

Warehouse(ID, Address, Manager, City†)
City(State, City, Headquarters)
Product(ItemID, Color, Description, Weight, Dimensions)
Order(OrderNum, Date, Name†)
Customer(Name, Address, Email)
ProductHolding(ID†, ItemID†, quantity)
ProductStoredIn(City†, ItemID†, Quantity)
ProductOrderedIn(ItemID†, OrderNum†, Quantity).