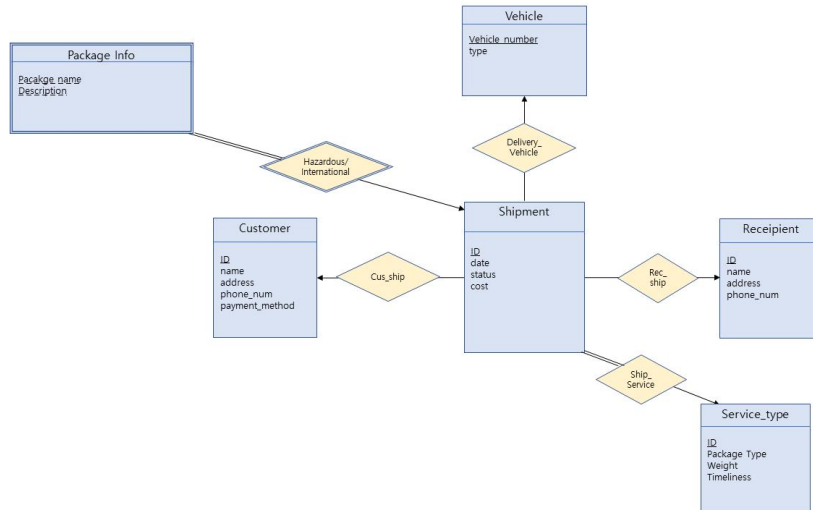


데이터 베이스 보고서

20191662 최찬우

[ER Model]



(Entity)

1. Customer Entity(고객)

ID: 고객이 회사의 서비스를 이용한 순간 부여되는 고유 번호(primary key)

name: 고객의 이름

address: 고객의 주소

phone_num: 고객의 핸드폰 번호

payment_method: 지불 방식에 대한 속성(account number 또는 credit card가 들어감)

2. Shipment Entity

ID: 배송이 되는 순간 부여되는 고유한 배송 번호(primary key)

date: 배송 날짜

status: 배송 상태(배송 중, 배송 완료(제시간), 배송 완료(지연) 등의 정보가 들어감)

cost: 현재 배송에 지불되는 비용

3. Receipt Entity

ID: 고객이 특정 수신자에게 배송하겠다고 서비스를 이용한 순간 수신자에게 고유 번호가 부여됨(primary key)

name: 수신자의 이름

address: 수신자의 주소

phone_num: 수신자의 휴대폰 번호

4. Vehicle Entity

Vehicle_number: 각 차량의 차량 번호(primary key)

type: 차량 종류(truck, plane 등)

5. Service_type

ID: 패키지의 여러 정보에 따라 이미 정해놓은 서비스 타입의 고유 번호(primary key)

Package Type: 포장지의 종류

Weight: 배송품의 무게

Timeliness: 배송 기한

6. Package Info

Package name: 배송 물품의 이름

Description: 배송 물품의 구체적 정보

(Entity 간 관계)

1. Customer - Shipment (one to many 관계)

고객은 여러 개의 배송을 요청할 수 있으며, 여러 고객이 하나의 동일한 배송 번호(운송 번호)를 가지는 경우는 존재할 수 없으므로 one to many 관계이며, 이를 Cus_ship으로 명칭하고, “배송 요청”이라고 해석한다.

2. Shipment - Receipient (many to one 관계)

여러 개의 배송 서비스가 동일한 수신자에게 도착할 수 있으며, 앞서 말했듯이 각각의 배송은 고객의 요청에 따라 고유한 배송번호가 부여되므로 한 개의 배송번호를 가지는 배송 서비스가 여러 명의 고객에게 도착하는 경우는 존재할 수 없으므로 many to one 관계이며, 이를 Rec_ship으로 명칭하고, “수신 예정”이라고 해석한다.

3. Shipment - Vehicle (many to one 관계)

배송 번호는 고객의 요청에 따라 개별적으로 부여되지만, 그렇다고 각각의 배송서비스 하나에 차량 하나를 이용하는 것은 비효율적이다. 따라서 여러개의 배송 서비스(배송 상품)를 하나의 운송 수단에 넣어서 배송을 진행하는 것이 더 바람직하다. 그러나 여기서 가정이 하나 존재하는데, 처음 출발하는 시점에 운행하는 운송수단이 배송 목적지까지 계속 해당 상품을 배달하고 있어야한다는 것이다. 따라서 이러한 케이스로 가정한다면 Shipment - Vehicle은 many to one 관계이며 이를 Delivery_Vehicle이라 명칭하고, “운행 차량”으로 해석한다.

4. Shipment - Service_type (many to one, Total & partial 관계)

여러 배송이 배송 상품의 상태에 따라 그에 대응하는 서비스 타입이 동일할 수도 있으며, 모든 배송은 반드시 그에 대응하는 서비스 타입이 존재해야한다. 따라서 두 관계는 many to one, Total & Partial 관계이며, 이를 Ship_Service로 명칭하고, “배송 서비스 타입”이라고 해석한다.

5. Shipment - Package Info (one to many , strong - weak entity set)

하나의 package에는 여러 가지 물건이 포장지에 포장되어 있는 상태일 것이다. 그리고 이러

한 package 하나가 운송 번호가 부여되어 배송이 된다. 이때 해당 package 안에 위험한 물건이나, 해외 배송 상품이 존재한다면 그 물건의 이름과 정보를 알아내야한다. 즉 하나의 배송에 대해 여러개의 물건이 나오게 될 수 있기 때문에 one to many 관계이다. 또한 Package Info는 무조건 배송 상품에 대해서만 확인하여 나오는 정보이기 때문에 Shipment Entity에 의존적인 Entity이므로 Weak Entity set이다. 이러한 관계를 Hazardous/International이라고 명칭하고, 위험 물품/해외 배송 물품 확인 이라고 해석한다.

(ER Model 테스트)

1. 트럭 1721번이 사고가 났을 경우 해당 트럭에 있는 package의 고객과 수신자의 정보를 알 수 있는가?

- Delivery_Vehicle set에서 Vehicle_number(1721번)을 조회하면 해당 트럭의 배송 운행 이력으로도 볼 수 있는 Shipment ID들을 알 수 있고, 여기서 얻은 모든 Shipment ID들 중에 현재 배송중인 status의 Shipment ID를 찾는다면, 고객과 수신자의 정보를 찾을 수 있다.

2. 위의 트럭이 사고 전 가장 최근에 배송에 성공한 정보를 얻을 수 있는가?

-1번과 동일하게 Delivery_Vehicle set에서 Vehicle_number(1721번)을 조회하고 여기서 얻은 모든 Shipment ID들을 이용해 가장 최근 date의 Shipment ID를 찾는다면 관련 정보를 얻을 수 있다.

3. 작년에 가장 많은 package를 배송한 고객을 찾을 수 있는가?

- Cus_ship set에는 Customer의 ID와 Shipment ID(primary key)가 저장되어 있다. 따라서 Cus_ship set에서 각 Customer ID별 Shipment ID들을 조회하고, 해당 Shipment ID를 이용해 Shipment set에서 date가 작년인것만 찾은 다음 이를 모든 고객에 대해 반복하여 가장 많은 package를 배송한 고객의 ID를 찾을 수 있다.

4. 작년에 가장 많은 돈을 소비한 고객을 찾을 수 있는가?

-3번과 동일하게 Cus_ship set에서 각 Customer ID별 Shipment ID들을 조회하고, 해당 Shipment ID를 이용해 Shipment set에서 date가 작년인것만 찾은 다음 그것들의 cost를 전부 더한다. 그리고 이를 모든 고객에 대해 반복하여 가장 비용이 큰 고객을 찾는다.

5. 제시 시간에 배송되지 않은 package들을 찾을 수 있는가?

-Shipment Set에서 status가 배송 완료(지연)과 관련된 정보를 찾으려면 된다.

6. 저번 달의 아래와 같은 상황에 대해 청구할 수 있는가?

6-1) Simple bill(고객 별 소비 비용)

-Cus_ship에서 Customer ID를 조회 후 나오는 Shipment ID를 이용하여, Shipment set에서 cost를 전부 조회하고 이를 더하여 bill을 청구할 수 있음

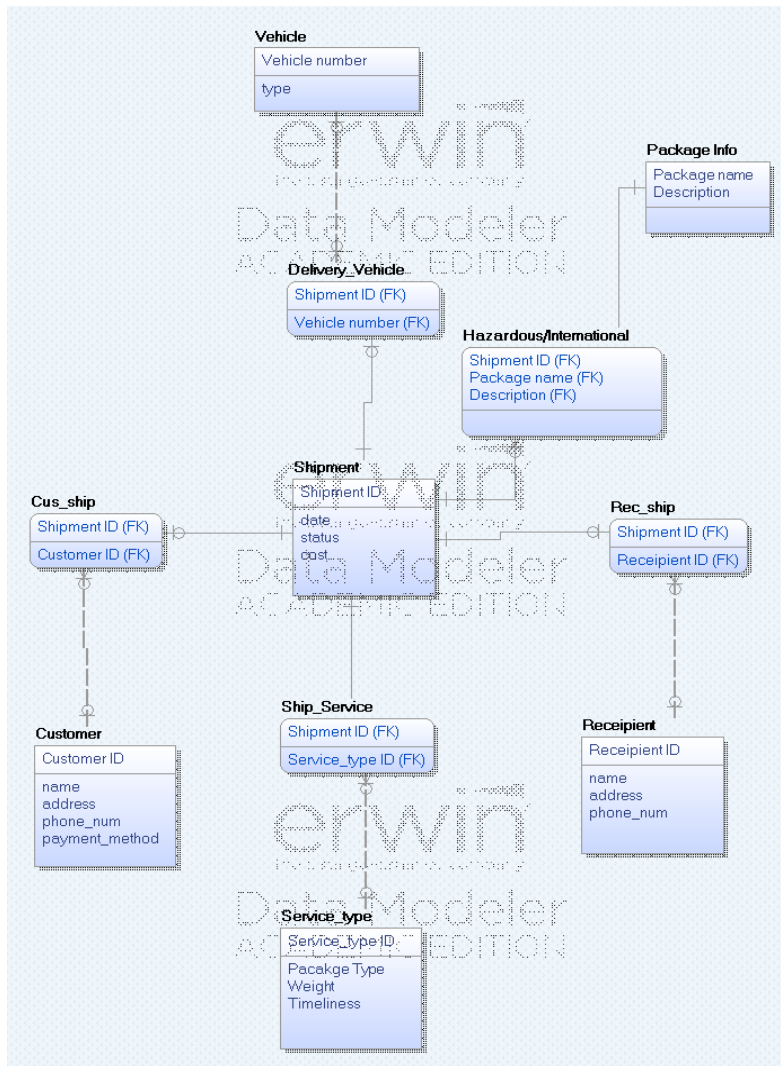
6-2) service type 별 비용

-Cus_ship에서 Customer ID를 조회 후 나오는 Shipment ID를 이용하여, Ship_Service set에서 각 Shipment ID에 대한 Service_type를 알아내면 Service type별 비용을 청구할 수 있음

6-3) 각 배송 별 비용

-Cus_ship에서 Customer ID를 조회 후 나오는 Shipment ID를 이용하여, Shipment set에서 각 Shipment ID 별 cost를 bill로 청구할 수 있음

(Relation Schema)



* ER 다이어그램에서의 relationship set들까지 전부 Schema Diagram에 포함시킨 이유는 Shipment Entity set과 관계가 있는 다른 Entity Set들이 많아서, Shipment Entity에 너무 많은 속성들이 포함될 수 있을 것 같아서 relationship set들도 전부 Schema Diagram에 포함시켰다.

1. Customer - Cus_ship (one to many)

한 고객 당 Cus_ship(배송 요청)을 여러 번 할 수 있기 때문에 두 set의 관계는 one to many이며, Customer와 Shipment는 one to many 관계이므로, Cus_ship의 foreign key로 Customer의 primary key가 들어온다.

2. Shipment - Cus_ship (one to one)

하나의 배송은 하나의 배송 요청으로부터 진행되므로 두 set의 관계는 one to one이며 Customer와 Shipment는 one to many 관계이므로, Cus_ship의 primary key에는 Shipment의 primary key가 들어온다.

3. Service type - Ship_Service (one to many)

하나의 서비스 타입은 여러개의 Ship_Service(배송 서비스 타입)에 포함될 수 있기 때문에 one to many 관계이며, Shipment와 Service_type은 many to one 관계이므로, Ship_Service의 foreign key에 Service type의 primary key가 들어온다.

4. Shipment - Ship_Service (one to one)

하나의 배송에는 하나의 배송 서비스 타입이 대응되어야 하므로 두 관계는 one to one이며 Shipment와 Service_type은 many to one 관계이므로, Ship_service의 primary key에 Shipment의 primary key가 들어온다.

5. Receipient - Rec_ship (one to many)

한 수신자 당 Rec_ship(수신 예정)인 상품이 여러 개일 수 있기 때문에 두 set의 관계는 one to many이며, Shipment와 Receipient는 many to one 관계이므로, Rec_ship의 foreign key로 Receipient의 primary key가 들어온다.

6. Shipment - Rec_ship (one to one)

하나의 배송은 하나의 수신 예정으로 이어지므로 두 set의 관계는 one to one이며 Shipment와 Receipient는 many to one 관계이므로, Rec_ship의 primary key에는 Shipment의 primary key가 들어온다.

7. Shipment - Delivery_Vehicle (one to one)

하나의 배송에 대해 하나의 Delivery_Vechicle(운행 차량)이 담당하므로, 두 관계는 one to one 이며, Shipment와 Vehicle은 one to many 관계이므로, Delivery_Vehicle의 primary key에는 Shipment의 primary key가 들어온다.

8. Vehicle - Delivery_Vehicle (one to many)

하나의 차량이 여러번의 배송을 담당할 수 있기 때문에 두 관계는 one to many 이며, Shipment와 Vehicle은 one to many 관계이므로, Delivery_Vehicle의 foreign key에 Vehicle의 primary key가 들어온다.

9. Shipment - Hazardous/International (one to many)

하나의 배송에 대해 위험물품 또는 국제 배송 물품이 여러 개 섞여 있을 수 있기 때문에, 두 관계는 one to many이며, Shipment와 Package Info는 one to many이며, Package Info set은 weak entity set이므로, Shipment의 primary key가 Hazardous/ International의 primary key로 들어온다.

10. Package Info - Hazardous/International (one to one)

Pacakge Info는 하나의 물품에 대해 위험 물품인지, 해외 배송 물품인지에 대해 설명하고 있으므로 Hazardous/International과 one to one 관계이며, Shipment와 Package Info는 one to many이며, Package Info set은 weak entity set이므로. Pacakge Info의 모든 key가