

DB Project 1

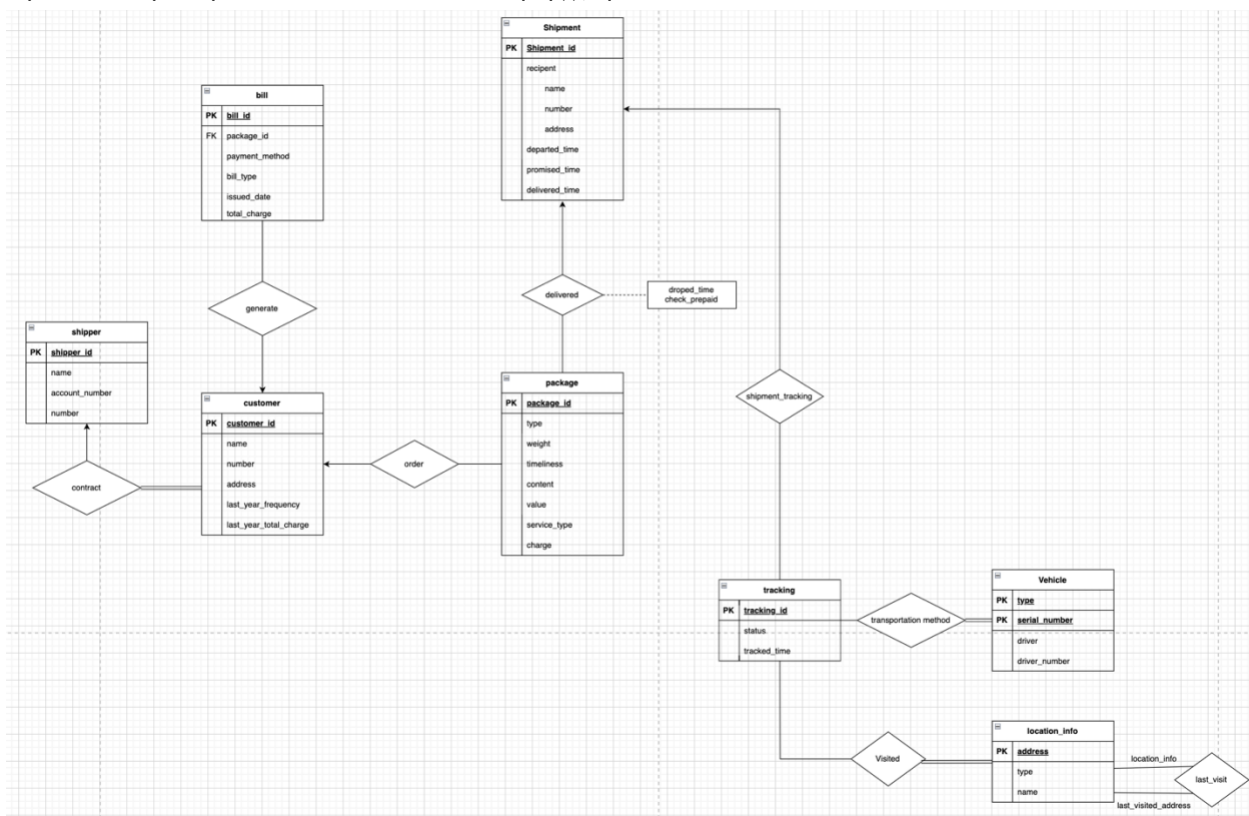
20191164 최윤지

1. 프로젝트 개요

E-R model 과 Schema Diagram 을 이용하여 Package Deliver System 을 위한 DBA 를 설계한다. 이때, 주어진 명세서의 조건과 Query 를 만족할 수 있도록 한다.

2. E-R Model

E-R model 설계할 때 가장 중요하게 고려한 점은 redundancy 가 일어나지 않도록 하는 것이다. 또한 자료의 수정, 삭제, 삽입이 일어날 때를 고려하여 새로운 entity 를 생성할지, 기존 Entity 의 attribute 로 정보를 삽입할지 등을 고려하였다. Primary Key 가 되는 attribute 가 많아져야 한다면 식별을 위한 고유 id 를 부여하여 Primary key 를 설정하는 등의 원칙을 기본으로 다음과 같은 ER model 을 설계하였다.

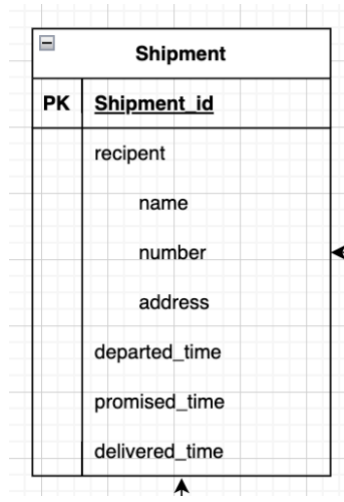


(Draw.io 를 이용하였으며, 다음 링크를 통해서도 방문할 수 있다.

https://drive.google.com/file/d/1SUqBqKSpwb5HMNGdLJXBUk_WCwckszqy/view?usp=sharing)

(1) Entity

Shipment



하나의 배송지로 향하는 shipment 에 대한 정보를 담은 entity 이다. Primary Key 로 shipment_id 를 갖는다. 수취인은 배송 중간에 변동되는 일이 없고 customer 로서 회사의 관리 대상이 아니기 때문에 따로 entity 를 지정하지 않고 shipment 의 attribute 로 작성하였다. Shipment 에는 하나가 아닌 여러 개의 다양한 종류, 보내는 사람 등이 다른 Package 들이 동시에 배달되므로 package 정보가 매 shipment 마다 복잡하게 연관이 된다고 판단하여 attribute 가 아닌 다른 entity 와 relation 으로 분류하였다. 배송 완료 예정 시간(promised_time)과 실제 완료 시간(delivered_time)을 비교할 수 있고 맨 처음 배송이 시작하는 출발 시간(departed_time)을 저장했다.

Package

package	
PK	<u>package_id</u>
	type
	weight
	timeliness
	content
	value
	service_type
	charge

하나의 물품에 관한 정보이다. Type 에 관한 정보를 저장하여 위험 품목인지 여부를 확인할 수 있다. 또한 weight, timeliness, content 등에 대한 정보를 가지고 있으며, 개별 가격(value) 와 배송 가격(charge)를 저장하도록 하였다. 물품 종류나 이름 등으로는 식별이 불가능하므로 primary key 로 package_id 를 지정하도록 했다.

Customer

customer	
PK	<u>customer_id</u>
	name
	number
	address
	last_year_frequency
	last_year_total_charge

물건을 주문하고 해당 물건에 대한 배송비를 지불하는 customer 에 관한 정보이다. 역시 동명이인 등이 있을 수 있으므로 customer_id 를 primary key 로 저장하였다. 작년에 배송한 물건의 총 수와 총 가격을 따로 저장해서 계산을 줄일 수 있도록 했다.

Shipper

shipper	
PK	<u>shipper_id</u>
	name
	account_number
	number

Customer 과 계약을 맺는 Shipper 에 관한 정보이다. 모든 customer 가 계약을 맺는 것이 아니고 shipper 에 대한 정보를 customer attribute 에 저장하는 건 null 값이 빈번하게 사용될 수 있다고 판단하여 entity 를 따로 만들어서 shipper 에 대한 정보를 저장하도록 했다. Shipper 는 account number 을 가지고 있으며, 이 attribute 는 계약을 맺은 customer 가 결제를 진행할 때 등의 Query 에 사용될 수 있다.

Bill

bill	
PK	<u>bill_id</u>
	package_id
	payment_method
	bill_type
	issued_date
	total_charge

Customer 가 배송한 물건의 배송료에 관한 bill 이다. 지불 수단과 발행일, 총 액수 등에 관한 정보를 저장한다. Bill_id 를 primary key 로 갖는다. 계약을 맺었는지 여부에 따라 지불 방법은 변할 수 있으나 지불을 하는 것은 변함이 없으므로 하나의 Entity 로 생성하였다.

Tracking

tracking	
PK	<u>tracking_id</u>
	status
	tracked_time

Shipment 의 운송 현황에 관한 정보이다. 배송 중인지, 배송이 완료되었는지, 준비중인지 등의 상태에 대한 정보를 저장하고 tracked 된 시간을 저장한다. Tracking id 를 갖는다.

Vehicle

Vehicle	
PK	<u>type</u>
PK	<u>serial number</u>
	driver
	driver_number

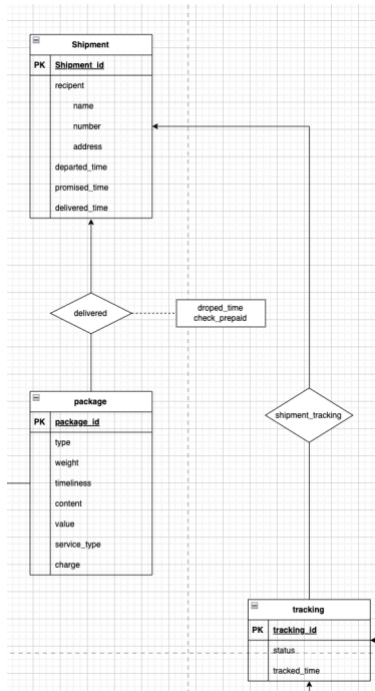
회사에서 관리하는 운송수단에 관한 정보이다. 비행기, 배, 트럭 등의 운송수단 종류와 시리얼 넘버로 primary key 를 갖는다. Driver 에 관한 정보를 저장하여 vehicle 에 문제가 생겼을 경우 연락할 수 있는 정보를 저장해둔다.

Location_info

location_info	
PK	<u>address</u>
	type
	name

운송 중 방문하게 되는 장소 등에 관한 정보이다. 물류 창고나 대리점 등에 관한 정보를 저장한다. 주소를 primary key 로 갖는다. 주소 등에 관한 검색이 용이하지 않을 경우 해당 이름이나 id 를 부여하여 primary key 로 지정할 수도 있으나 관한 내용이 세부적으로 설명되지 않아 주소로 검색 등이 용이하다고 가정하고 판단하였다. Warehouse 에 관한 세부적인 내용이 명세서에서 주어지지 않아 임의로 설정하였으나 만약 같은 이름을 가지고 있는 warehouse 가 없다면 이름을 primary key 로 저장할 수도 있다.

(2) Relationship

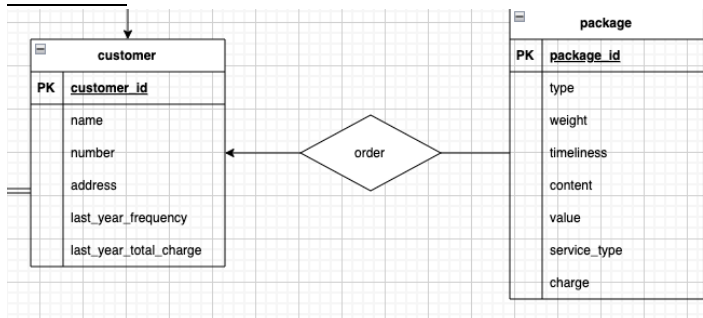


(shipment_tracking 과 delivered 가 함께 나타난 부분)

Delivered

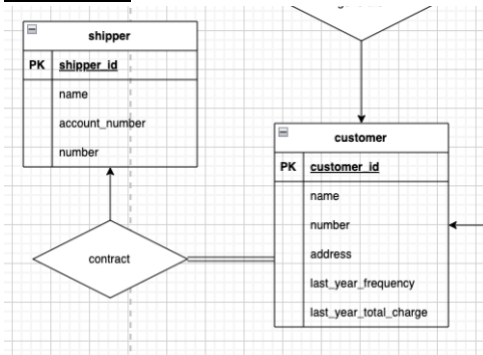
Shipment 와 package 사이의 관계로, 하나의 Shipment 에는 여러개의 package 가 함께 배송되지만 한 package 는 여러 번 배송되지 않으므로 one to many 의 관계를 가진다. 추가적인 속성으로 해당 패키지에 대한 배송비가 미리 지불되었는지, 패키지가 언제 shipment 를 위해 전달되었는지(dropped_time)를 함께 저장한다.

Ordered



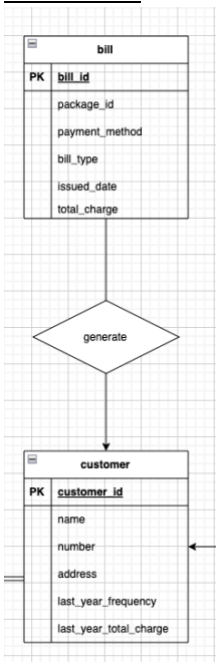
Package 와 customer 가 갖는 관계로, 하나의 package 는 오직 한 번(하나의 customer)만 구매되지만 customer 는 여러 개의 package 를 구매할 수 있으므로 many to one 의 관계를 갖는다. 추가 속성으로 Ordered_time 등을 줄 수도 있다고 생각했으나 택배회사 입장에서 해당 정보는 그다지 중요하지 않다고 생각하여 포함하지 않았다.

Contract



Shipper 과 customer 가 갖는 관계이다. Customer 는 한 명의 shipper 와 계약을 맺지만 shipper 은 여러 명의 customer 과 계약을 맺을 수 있게끔 one to many 관계로 설계하였다. 또한 모든 customer 가 계약을 맺는 것이 아니므로 partial 관계이다.

Generate

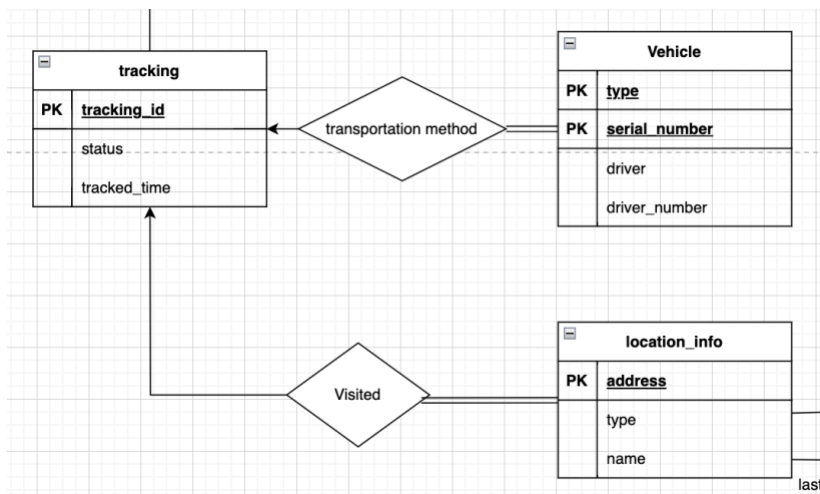


Customer 와 Bill 의 관계이다. 하나의 customer 는 bill 을 여러번 Generate 할 수 있지만, 하나의 bill 은 하나의 customer 에게만 할당된다. 따라서 one to many 관계를 가진다.

Shipment_tracking

((2)의 맨 첫번째 사진 참조)

Tracking 과 shipment 에 관한 관계이다. Shipment 는 한번에 배송되지 않고 여러 번의 tracking 과정을 거치지만 tracking 은 하나의 shipment 에만 할당된 정보이므로 many to one 의 관계를 갖는다.



(transportation method 와 visited relation 이 함께 나타난 부분)

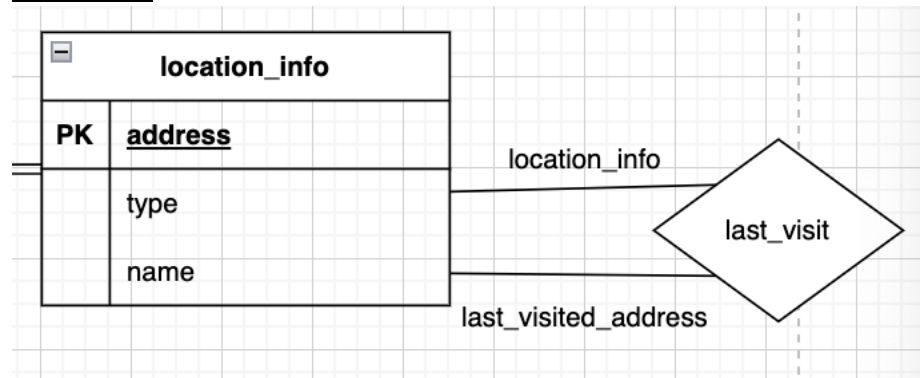
Transportation_method

Vehicle 과 tracking 간의 관계이다. Vehicle 은 여러 번의 tracking 에 관여할 수 있지만 tracking 이 동시에 여러 대의 vehicle 을 동원할 수 없으므로 one to many 관계이다. 또한 어떤 vehicle 은 tracking 에 관여하지 않을 수도 있으므로 partial 관계를 갖는다.

Visited

Tracking 과정 중 방문한 현재 위치에 대한 정보이다. Vehicle 과 마찬가지로 여러 곳을 경유해서 도착지까지 배송되기 때문에 one to many 의 관계를 갖는다.

Last_visit



Role 관계를 수행한다. 이전에 방문한 위치를 정할 수 있도록 하여 user 나 회사에서 배송 경로를 추적할 때 이용할 수 있도록 한다. Tracking 정보를 표현하는 과정에서 고민이 있었으나 상용 배송 추적 시스템 등에서 배송 기사 연락처 등의 정보를 함께 보여주는 점, 이전 방문지(물류창고, hub 등)는 확인할 수 있으나 다음 방문지는 미리 알려주지 않는다는 점을 착안하여 설계하였다.

3. Schema Relationship

위의 E-R model 를 이용, Erwin program 을 통해 작성한 schema diagram 은 다음과 같다.

