Project TDT4145

Delivery part 1: ER-model and relational schema

Emil Jønsrud, Karen Pernille Ljosland Karlsen, Sigrid Maria Nese

ER-model of the complete data model

See: Appendix A: ER-model of the complete data model

Comments and assumptions

We have assumed that all operators have the same customer registry. Otherwise, we would make customers weak entities to a CustomerRegistry-entity. Hence, the operators' common registry is captured in the Customer entity.

All primary keys can not have NULL-values.

We assume that all TrackSubSections need to be part of a TrainRoute. Therefore, TrackSubSection has a (1, n) relation to TrainRoute.

A CarInTrain is either a ChairCar or a SleepCar, and nothing else. These entities do not share attributes. Therefore, this specialization is total and disjoint.

We have assumed that all customers *must* register with a name, email and a phone number. Notice that we only store one phone number, as it does not make sense with multiple in this case.

Several of the relations have cardinality (1, 1) from one of the entities and (0, n) from the other. In cases where it could be more logical with (1, n). We have however, chosen to use cardinality (0, n), as the implementation would not make sense if both entities need to exist before the other can exist. This also holds for (1,n)-(1,n) relation.

We assume that the TrackSections are independent, and do not include any other intersecting TrackSections (apart from junction points).

The information describing electric/diesel driving-energy and single/double tracks are defined as boolean attributes. This is because it is either one or the other.

Relational schemas

TrackSection(Name, ElectricTracks, *StartStation, EndStation*)

- Name is primary key.
- StartStation is foreign key to the RailwayStation-table. Can not have NULL-value.
- EndStation is foreign key to the RailwayStation-table. Can not have NULL-value.

RailwayStation(Name, Height)

- Name is primary key.

TrackSubSection(SectionNo, *SectionName,* Distance, DoubleTrack, *StartsAt*, *EndsAt*)

- SectionNo and SectionName are primary keys.
- SectionName is foreign key to the TrackSection-table. Can not have NULL-value
- StartsAt is foreign key to the RailWaystation-table. Can not have NULL-value.
- EndsAt is foreign key to the RailWaystation-table. Can not have NULL-value.

TrainRoute(Name, Operator, *TrackName*)

- Name is primary key.
- TrackName is foreign key to the TrackSection table. Can not have NULL-value.

WeekDay(Name)

- Name is primary key.

DayOfRoute(NameOfDay, NameOfRoute)

- NameOfDay and NameOfRoute are primary keys
- NameOfDay is foreign key to the WeekDay-table. Can not have NULL-values.
- NameOfRoute is foreign key to the TrainRoute-table. Can not have NULL-values.

RouteStop(*Station*, *NameOfRoute*, TimeOfDay)

- Station and NameOfRoute are both primary keys.
- Station is foreign key to the RailwayStation-table.
- NameOfRoute is foreign key to the TrainRoute-table

TrainOccurance(RunDate, *NameOfRoute*)

- RunDate and NameOfRoute are primary keys.
- NameOfRoute is foreign key to the TrainRoute-table. Can not have NULL-value.

Placement(PlaceNo, *CarID*)

- PlaceNo and CarID are primary keys.
- CarID is foreign key to CarInTrain.

CarInTrain(CarID, CarNo)

- CarID is primary key.

ChairCar(CarID, NumOfRows, SeatsPerRow)

- CarID is primary key and foreign key to the CarInTrain-table.

SleepCar(CarID, NumOfCompartments)

- CarID is primary key and foreign key to the CarInTrain-table.

Ticket(TicketNo, *OrderID*, *CarID*, *PlaceNo*)

- TicketNo and OrderID are primary keys.
- OrderID is foreign key to CustomerOrder. Can not have NULL-value.
- PlaceNo and CarID are foreign keys to the Placement-table. Can not have NULL-value.

CustomerOrder(OrderID, OrderDate, OrderTime, *CustomerID*)

- OrderID is primary key.
- CustomerID is foreign key to the Customer-table. Can not have NULL-value.

Customer(CustomerID, Name, Email, PhoneNo)

- CustomerID is primary key.

All entities are on the fourth normal form, as their primary key is unique and does not depend on other attributes. This comes from the fact that anything can be functionally dependent of a unique ID. For example a CustomerOrder can have several orders for the same CustomerID, however different Customers cannot have the same OrderID.

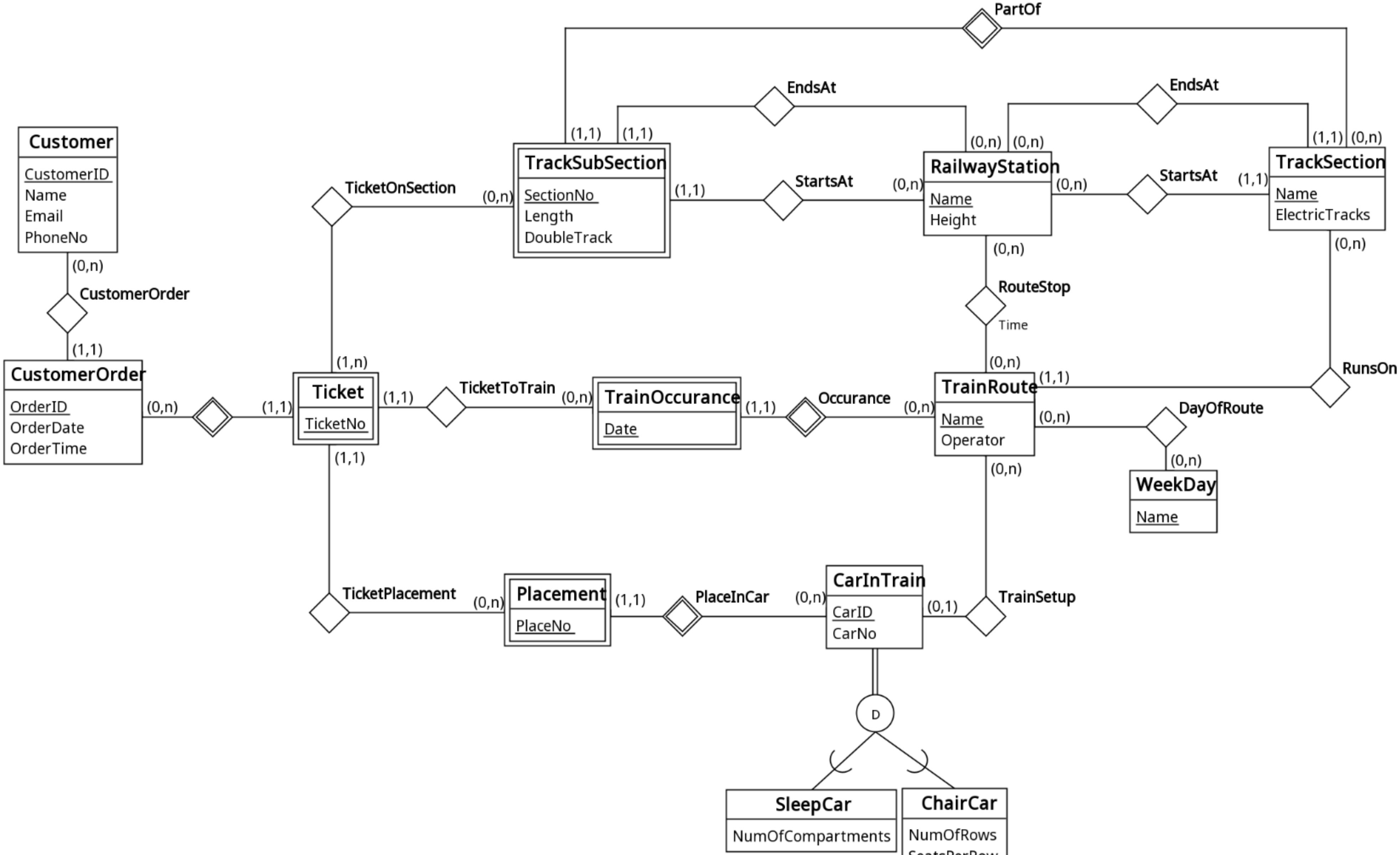
The SQL script

Please see the attached sql-script for generating the relational schemas.

Required application functionality

When a bed in a coupe is booked for only a portion of a TrainOccurance, the coupe must be unavailable to book for the entire TrainOccurance. This functionality needs to be implemented in the application, as this behavior is not defined in the relational schema.
Similarly, the relational schema opens for multiple tickets to be sold to the same seat on the same sections of the route. This also needs to be accounted for in the application.
The reasoning for relating a TrainRoute to a WeekDay is to open up for automatic generation of TrainOccurances. This can be done in the application.

Appendix A: ER-model of the complete data model


PartOf
EndsAt
EndsAt
(1,1)
(1,1)
(0,n)
(0,n)
(1,1)
(0,n)
Customer
CustomerID
Name
Email
PhoneNo
TicketOnSection
(0,n)
TrackSubSection
SectionNo
Length
DoubleTrack
(1,1)
StartsAt
(0,n)
RailwayStation
Name
Height
(0,n)
StartsAt
(1,1)
TrackSection
Name
ElectricTracks
(0,n)
(0,n)
CustomerOrder
(1,1)
(0,n)
RouteStop
Time
(0,n)
RunsOn
(1,n)
CustomerOrder
OrderID
OrderDate
OrderTime
(0,n)
(1,1)
Ticket
TicketNo
(1,1)
TicketToTrain
(0,n)
TrainOccurance
Date
(1,1)
Occurance
(0,n)
TrainRoute
Name
Operator
(1,1)
(0,n)
DayOfRoute
(0,n)
WeekDay
Name
(1,1)
(0,n)
TicketPlacement
(0,n)
Placement
PlaceNo
(1,1)
PlaceInCar
(0,n)
CarInTrain
CarID
CarNo
(0,1)
TrainSetup
D
SleepCar
NumOfCompartments
ChairCar
NumOfRows
SeatsPerRow