TDT4145 Project 2023 - Delivery 1

Students:

Last name	First name	Email
Olsen	Andreas Omholt	andreaoo@stud.ntnu.no
Punnerud	Jakob Eilertsen	jakobep@stud.ntnu.no
Ringheim	Iver	iverri@stud.ntnu.no

Table of contents

ER-model

Below you will find an ER-diagram modeling the mini world describing our database over the Norwegian railway system (figure 1). The given mini world is quite complex, and it would be difficult to make a good model without the use of enhanced entity-relationship (EER) modeling. We have used multiple concepts from both ordinary and enhanced ER. This includes ordinary entities, weak entities, binary relations, identifying relations and total and disjoint specialization. Even though this is an EER-model, we will refer to the model as an ER-model.

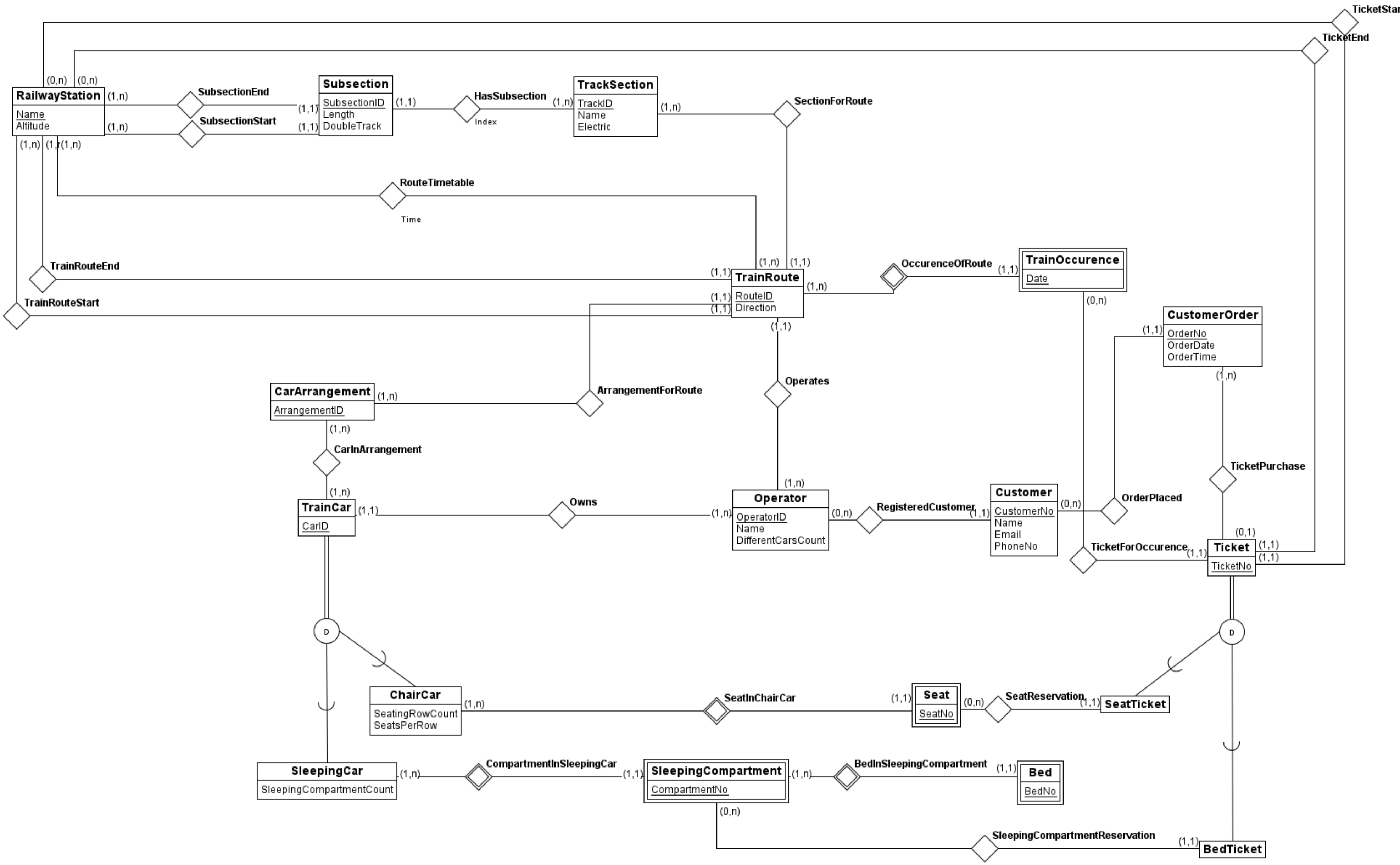

Figure 1: ER-diagram for the Norwegian railway system. (Please consult the attached PNG if it is difficult to see properly.)

Assumptions and constraints for ER-model

There are no railway stations with the same name, hence the primary key Name in the RailwayStation entity.

Assumes that a subsection of a physical track section is solely the stretch between two railway stations with no other stations in between. As a result of this we have the relations SubsectionStart and SubsectionEnd between the Subsection entity and RailwayStation entity.

Assumes that a train route is operated with the same car arrangement every time. By this logic we have the relation ArrangementForRoute with cardinality (1,1) from the TrainRoute point of view. We also assume that a car arrangement can be used on multiple routes. Additionally we assume that every railroad car must be included in a CarArrangement, and that any CarArrangement must include at least one car. Therefore we have (1,n) cardinalities on both sides of the CarInArrangement relation between CarArrangement and TrainCar.

Assumes that a customer can be registered in the operators registry without any placed orders. Thus the cardinality (0,n) on the Customer side of the OrderPlaced relation between Customer and CustomerOrder.

TrainCar is a total and disjoint specialization since a car cannot be both a seating car and a sleeping car at the same time, and a car must be one of the specified types. Ticket is a total and disjoint specialization since a ticket either reserves a seat or a sleeping compartment.

Assumes that an order must include at least one ticket purchase. Hence the cardinality (1, n) on the CustomerOrder side of the TicketPurchase relation between Ticket and CustomerOrder. We also assume that a ticket can exist before it has been bought. I.e. it is not part of a customer order. Therefore the cardinality (0, 1) on the Ticket side of the TicketPurchase relation between Ticket and CustomerOrder.

Assumes that each train route is only operated, at most once, in a twenty-four hour period. Therefore we can uniquely identify a TrainOccurence with the partial key Date using the identifying relation OccurenceOfRoute. This relation is between TrainRoute and TrainOccurence.

Additionally, the constraint of not being able to sell a ticket to a bed if it already has been occupied on the same train occurrence is too difficult to represent in the ER-model.

Relational tables and normal forms

When mapping ER-models to relational database schemas it is important to normalize the tables to avoid redundancies and anomalies. Normal forms are a set of rules a table must fulfill. A table may achieve a higher normal form by being decomposed.

Tables in the first normal form (1NF) are characterized by having atomic attributes, that means that an attribute is a single value from the given domain (Midtstraum, 2021a). 1NF tables do not have composite attributes or duplicate rows. All of the higher normal forms presuppose 1NF.

A table is in the second normal form (2NF) if there are no non-key attributes that are partially dependent on a composite candidate key (Midtstraum, 2021a).

A table is in the third normal form (3NF) if and only if for all functional dependencies on the form X -> A (Midtstraum, 2021a), one of the following rules must apply:

- X is a super key for the table
- A is a key attribute

A table is in Boyce-Codd normal form (BCNF) if for all functional dependencies on the form X -> A, X is a super key for the table (Midtstraum, 2021a). That means that all determinants on the left side of a functional dependency is a super key.

A table is in the fourth normal form (4NF) if for all non-trivial multivalued dependencies on the form X ->> A, X is a super key for the table (Midtstraum, 2021b).

As none of our tables in the relational schema defies the demands for 4NF, all tables are in 4NF.

- RailwayStation(Name, Altitude)
    - Primary key: Name

- Subsection(SubsectionID, Length, Double Track, SubsectionStart, SubsectionEnd)
    - Primary key: SubsectionID
    - Foreign key: StartStation references RailwayStation(Name)
    - Foreign key: EndStation references RailwayStation(Name)

- TrackSection(TrackID, Name, Electric, TrackSectionStart, TrackSectionEnd)
    - Primary key: TrackID
    - Foreign key: TrackSectionStart references RailwayStation(Name)
    - Foreign key: TrackSectionEnd references RailwayStation(Name)

- HasSubsection(SubsectionID, TrackID, Index)
    - Primary key: (SubsectionID, TrackID)
    - Foreign key: SubsectionID references Subsection(SubsectionID)
    - Foreign key: TrackID references TrackSection(TrackID)

- TrainRoute(RouteID, MainDirection, OperatorID, TrackID, ArrangementID)
    - Primary key: TrainRouteID

- Foreign key: OperatorID references Operator(OperatorID)
- Foreign key: TrackID references TrackSection(TrackID)
- Foreign key: ArrangementID references CarArrangement(ArrangementID)

- TrainOccurence(RouteID, Date)
  - Primary key: (RouteID, Date)
  - Foreign key: RouteID references TrainRoute(RouteID)

- Customer(CustomerNo, Name, Email, PhoneNo, OperatorID)
  - Primary key: CustomerNo
  - Foreign key: OperatorID references Operator(OperatorID)

- Operator(OperatorID, Name, DifferentCarsCount)
  - Primary key: OperatorID

- CustomerOrder(OrderNo, OrderDate, OrderTime)
  - Primary key: OrderNo
  - Foreign key: CustomerNo references Customer(CustomerNo))

- SeatTicket(TicketNo, OrderNo, TicketStart, TicketEnd, RouteID, Date, CarID, SeatNo)
  - Primary key: TicketNo
  - Foreign key: OrderNo references CustomerOrder(OrderNo)
  - Foreign key: TicketStart references RailwayStation(Name)
  - Foreign key: TicketEnd references RailwayStation(Name)
  - Foreign key: (RouteID, Date) references TrainOccurence(RouteID, Date)
  - Foreign key: (CarID, SeatNo) references Seat(CarID, SeatNo)

- BedTicket(TicketNo, OrderNo, TicketStart, TicketEnd, RouteID, Date, CarID, BedNo)
  - Primary key: TicketNo
  - Foreign key: OrderNo references CustomerOrder(OrderNo)

- Foreign key: ticketStart references RailwayStation(Name)
- Foreign key: ticketEnd references RailwayStation(Name)
- Foreign key: (RouteID, Date) references TrainOccurence(RouteID, Date)
- Foreign key: (CarID, BedNo) references Bed(CarID, BedNo)

- Seat(<u>CarID</u>, <u>SeatNumber</u>)
  - Primary key: (CarID, SeatNumber)
  - Foreign key: CarID references ChairCar(CarID)

- Car(<u>CarID</u>, OperatorID)
  - Primary key: CarID
  - Foreign key: OperatorID references Operator(OperatorID)

- Bed(<u>CarID</u>, <u>BedNo</u>)
  - Primary key (<u>CarID</u>, <u>BedNo</u>)
  - Foreign key: CarID references SleepingCar(CarID)

- ChairCar(<u>CarID</u>, SeatingRowCount, SeatsPerRow, OperatorID)
  - Foreign key: CarID references Car(CarID)

- SleepingCar(<u>CarID</u>, SleepingCompartmentCount, OperatorID)
  - Foreign key: CarID
  - Foreign key: OperatorID references Operator(OpeatorID)

- CarArrangement(<u>ArrangementID</u>)
  - Primary key: ArrangementID

- CarInArrangement(<u>ArrangementID</u>, <u>CarID)</u>
  - Primary key: (ArrangementID, CarID)
  - Foreign key: ArrangementID references CarArrangement
  - Foreign key: CarID references ChairCar(CarID) or SleepingCar(CarID)

- RouteTimetable(RouteID, Station, TimeOfDay)
    - Primary key: (RouteID, RailwayStation)
    - Foreign key: RouteID references TrainRoute(RouteID)
    - Foreign key: Station references RailwayStation(Name)

SQL script and application restrictions

In the SQL script almost all columns in all tables have a NOT NULL restriction. The sole exception is the OrderNo column in the SeatTicket and BedTicket tables. This is because from our assumptions a ticket exists before it is part of an order, and an order can be canceled without removing the ticket entity. This means that a customer can order a ticket that another customer has canceled. We also contemplated not having a NOT NULL restriction for CustomerID in CustomerOrder. This would keep the CustomerOrder saved in the database even if the Customer removed its registration from the database (by deleting their profile for example). This could end up useful for the Operator with regard to data analysis purposes. We ended up keeping the NOT NULL restriction because we felt our basis for removing it relied too much on speculation about what the Operator would use the database for. It was also not brought up in the problem description or in any of the user stories.

Another restriction in the SQL script is imposed on the RouteDate attribute. This is the UNIQUE restriction which ensures that there are not two train occurrences on the same day. This is to stay consistent with our assumptions and also the problem description. All tables also have a primary key to ensure unique instances of a relation.

Some restrictions have to be implemented in the application program. The fact that a bed ticket also reserves the entire compartment. This has not been implemented in the SQL script. This is because we could not find any intuitive way of implementing this, without reducing the resulting table to 2NF instead of 4NF. Additionally, this restriction would be much easier to implement in the application program with if-statements so we thought it reasonable to not implement it in the SQL script.

References

Midtstraum, R. (2021a, 16. februar). Normalformer fra 1NF til BCNF [PowerPoint-lysbilder]. Blackboard NTNU.

Midtstraum, R. (2021b, 16. februar). Normaliseringsteori: MVD-er, 4NF [PowerPoint-lysbilder]. Blackboard NTNU.