

This is an exercise from the Merode course at KU Leuven.

The solution is given as an EDG (part of a Merode model)

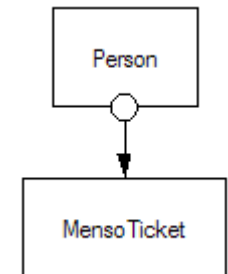
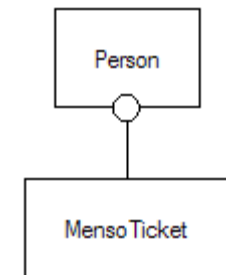
Some of the mistakes from student solutions are specific to Merode, others are universal

EDG

MENSO SEASON TICKETS

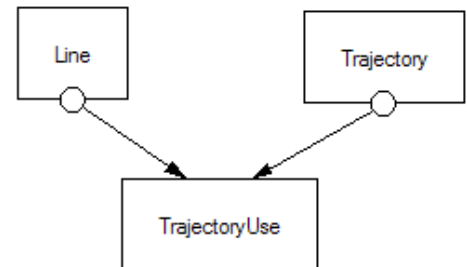
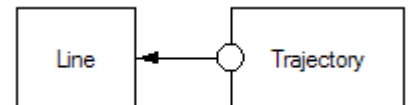
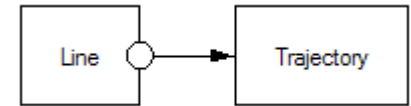
Person versus Menso Ticket

- Consider whether you want to make a distinction between a person and his/her Menso Ticket
- If you decide to make distinction, how many tickets a person have at one point in time ?
 - If only one, this implies that all information related to the “old” ticket must be ended (\neq deleted) before a new ticket can be issued. This could mean, for example, that all invoices have to be paid before a new ticket can be issued.
 - If many, consider whether a person is able to have to have several valid tickets in use at one point in time. If not, consider putting the old ticket on hold before issuing the new ticket
- *Note that the word "ticket" is misleading. Maybe "subscription" or "badge" would be a better name, as many trips can be done with one MensoTicket/Subscription/Badge.*



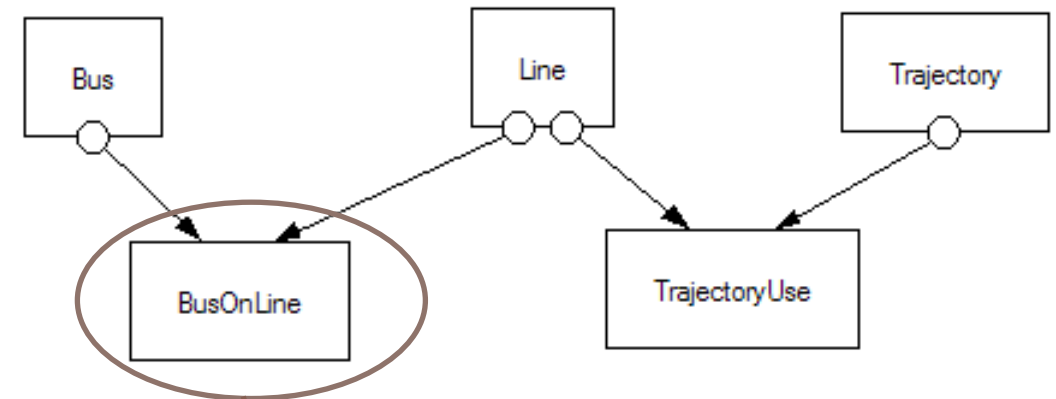
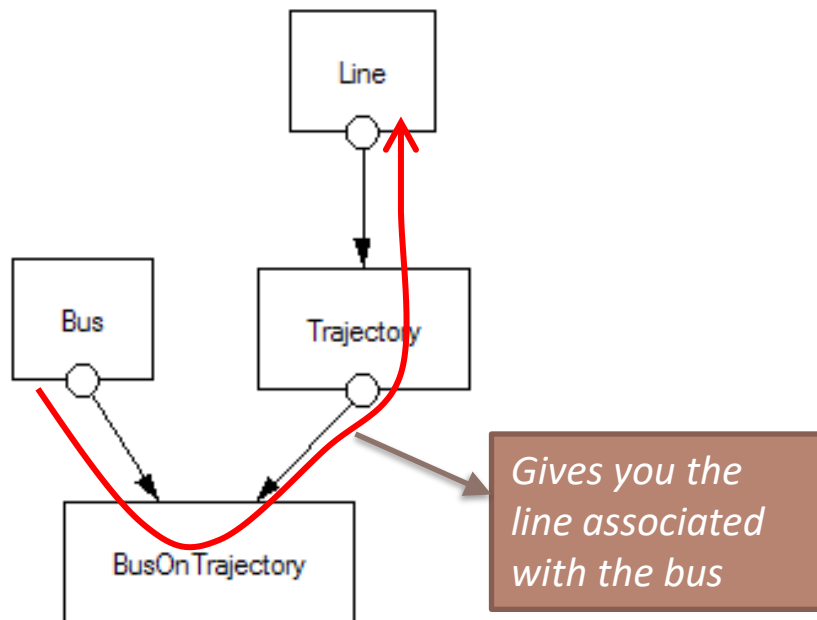
Trajectory - Line

- What is the definition of a trajectory versus the definition of a line ? Is it the same or are they different concepts ?
 - According to the case description we face this situation:
 - A line groups several overlapping trajectories. Hence, the line exists first, and one or more trajectories are defined per Line. (E.g. in Leuven, one has Line 2 Campus and Line 2 Boskant which follow partly the same and partly different routes.)
- Other options can be considered (in different countries/cities...)
 - A trajectory goes from A to B following a particular route. To this trajectory one (or many ?) lines can be assigned.
 - A trajectory goes from A to B following a particular route. A line is a composition of trajectories. Trajectories can be shared across lines.



Bus – assignment to line/trajectory

- How does one assign buses to lines or trajectories ?
- It will depend on how these two object types are connected.
- In any case, consider assigning the bus to the “lowest object” in the EDG, as the master can be navigated to following the associations between classes.

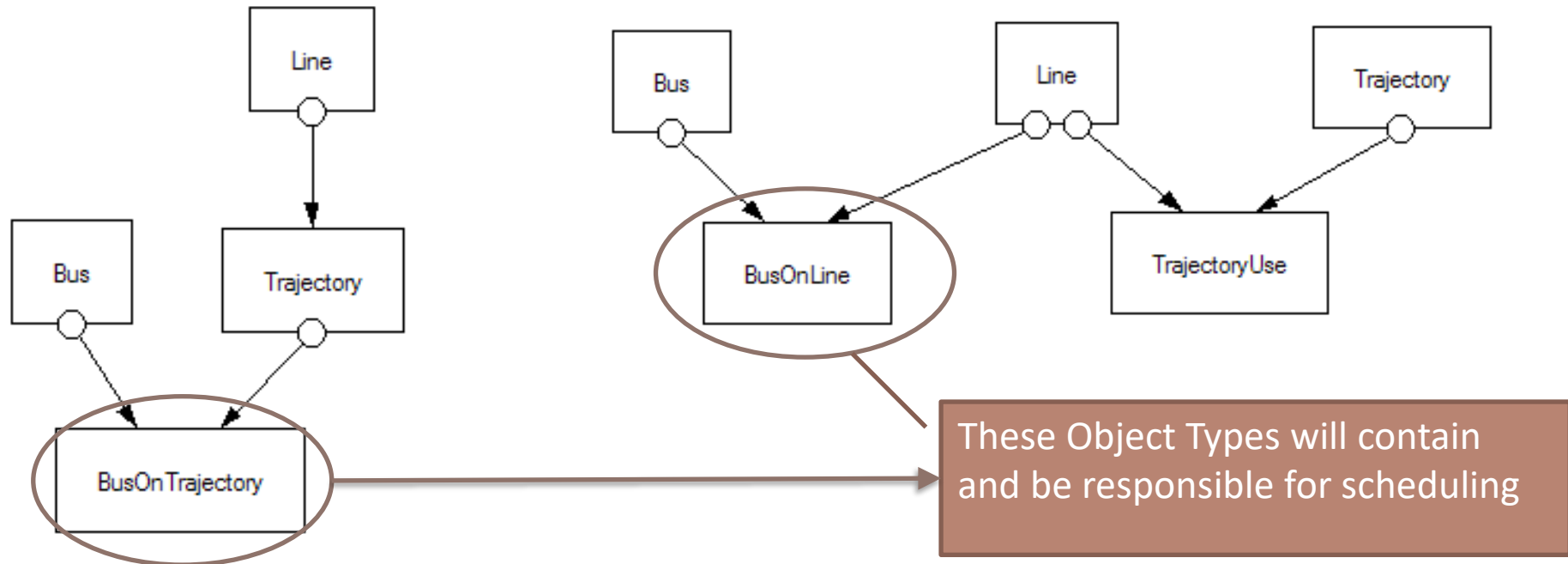


This could be the way to go if you want to be able to change the trajectory of a line without having to re-assign the busses:

- Assigning busses and Assigning the trajectory are two actions that can be done independently from each other.

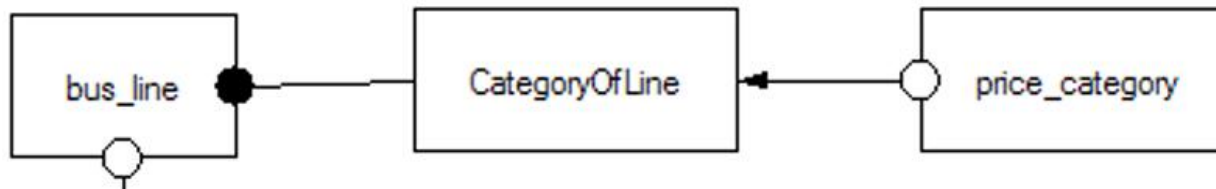
Bus – assignment to line/trajectory

- Cardinality ? How many lines do you associate a bus to ? Only one is the easiest. ‘Many’ enables flexible scheduling: depending on the time of the day and the needs on a particular line, a bus will follow different trajectories according to its schedule. Scheduling information should be managed so as to ensure that overlapping assignments are avoided as a bus cannot be driving two different routes at the same point in time ...



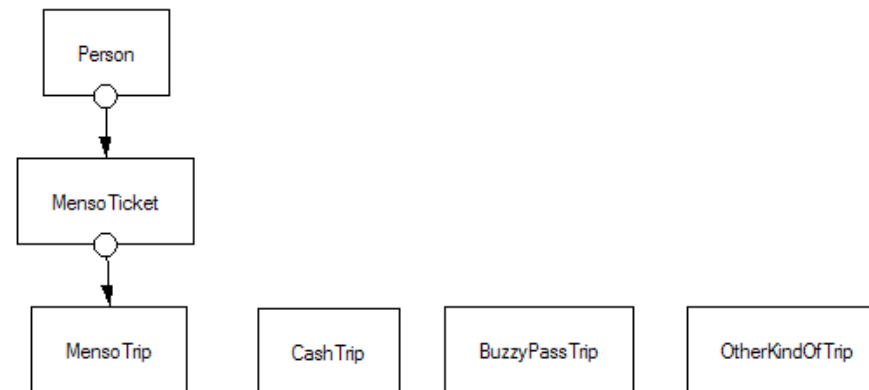
Category

- Is this an object type or an attribute ? Connected to which object type ? Trajectory or Line ?
- Can the category of a Line/Trajectory be changed over time
 - if no → ED-relationship can be used
 - if yes → use an intermediate class
- Assumption made:
 - A Category (PriceCategory) is an object type. One of its attributes will be "price".
 - It exists on its own (not ED of BusLine), and a BusLine can be assigned a different PriceCategory over time. But it must have one at any point in time.



Trip

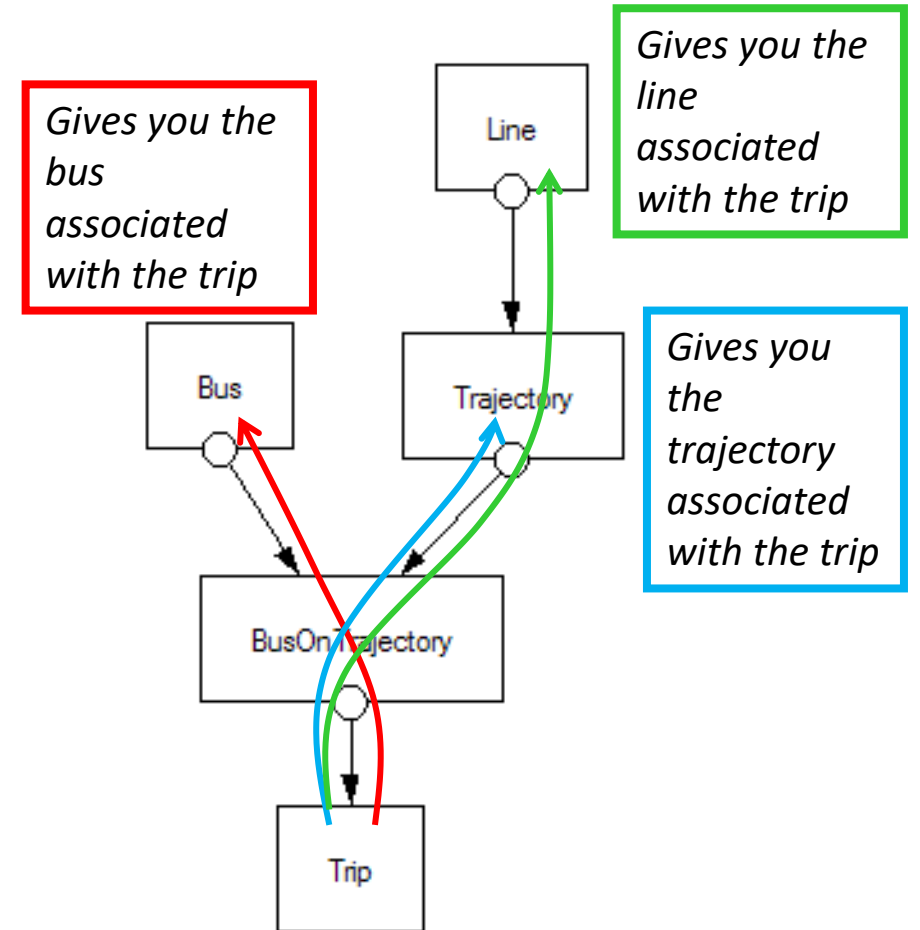
- Can a trip be anonymous or not
 - Every trip is made by a customer. But is it important for the company to know the customer ? For all kinds of trips ?
 - ➔ a business model models the real world *from the perspective of a business*. This means you can choose to leave out certain aspects of the real world.



- How many trips are there for a customer at one point in time ? Depends on when you consider a trip to be ended : when the customer gets off the bus or when the trip has been invoiced and paid ?
- Different kinds of trips can be modelled using inheritance, but this is postponed to chapter 8 ...

Trip

- What do you want to know about the travel medium:
 - the bus ?
 - the line ?
 - the trajectory ?
- Generally speaking, connecting to the most existence dependent object gives you the most information, as the information contained in the master object types can be found by navigating the associations



Invoice

- Who do you send the invoice to ?
 - to the customer
 - to the customer of the menso ticket
- ➔ in the latter case, connect the invoice to the ticket
- Every invoice line invoices a trip
 - The trip is made first, the invoice line is created afterwards
 - ➔ Trip ≠ InvoiceLine

Template Solution

