SE SOFTWARE ENGINEERING

RWTH AACHEN

Exercise

Object diagrams

Prof. Dr. Bernhard Rumpe
Lehrstuhl für Software Engineering
RWTH Aachen

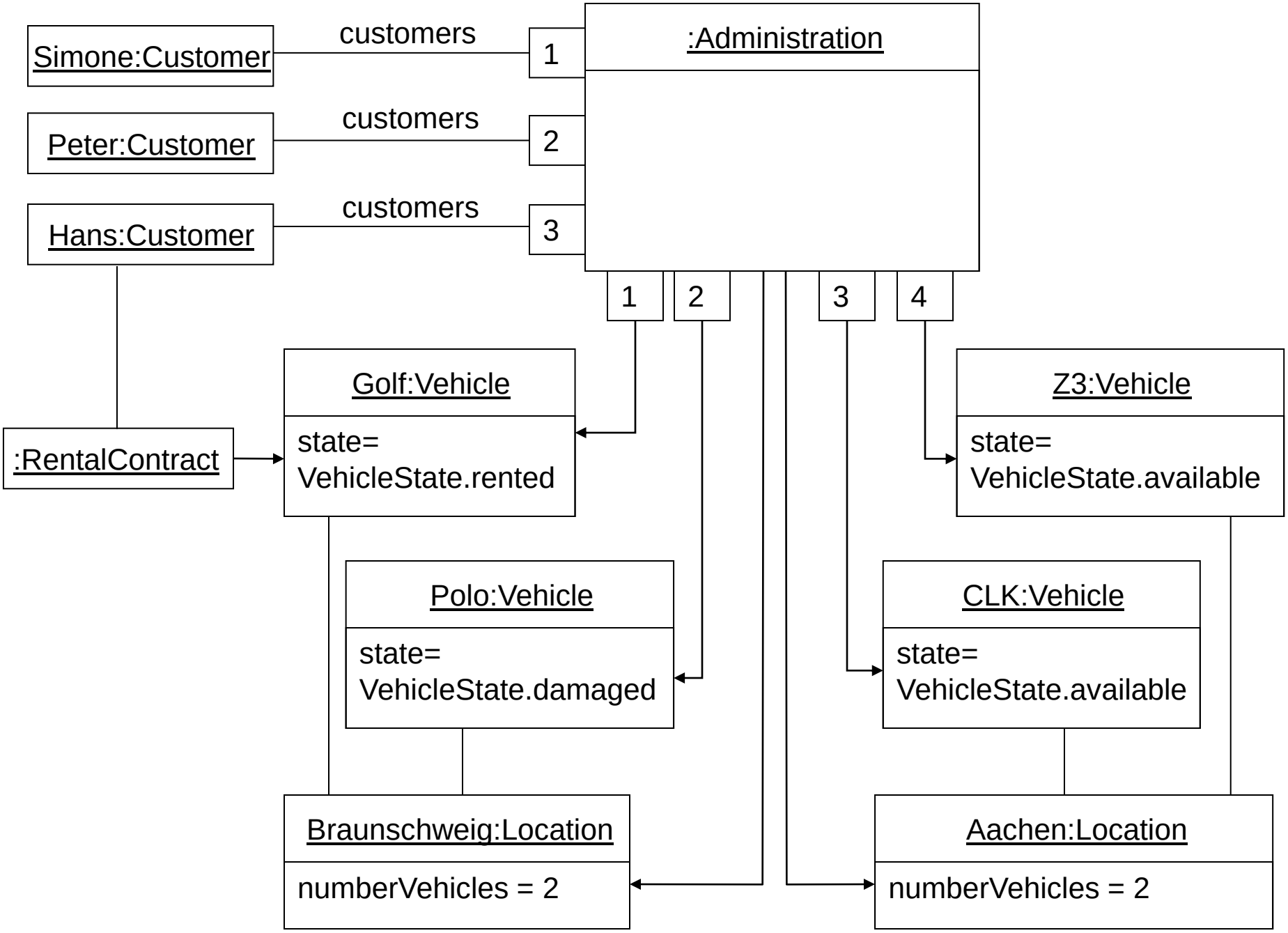
http://www.se-rwth.de/

Task 6.1 - Create an Object Diagram

Model an object diagram from the following description. Attributes, whose values are not called in the setting of task, can be omitted.

- *The car hire possesses the persons Hans, Peter and Simone as customers. The vehicle park contains a BMW Z3, a Mercedes CLK, a VW Polo, and a VW Golf.*

- *The customer IDs and vehicle IDs are consecutively numbered. The car hire has two locations in Braunschweig and Aachen. The Aachen location possesses the BMW Z3 and the Mercedes CLK. Both cars are available. At the Braunschweiger location the Polo and the Golf are present. The Polo is damaged and the Golf is rented to Hans.*


Simone:Customer
customers
1
:Administration
Peter:Customer
customers
2
Hans:Customer
customers
3
1
2
3
4
Golf:Vehicle
state=
VehicleState.rented
Z3:Vehicle
state=
VehicleState.available
:RentalContract
Polo:Vehicle
state=
VehicleState.damaged
CLK:Vehicle
state=
VehicleState.available
Braunschweig:Location
numberVehicles = 2
Aachen:Location
numberVehicles = 2

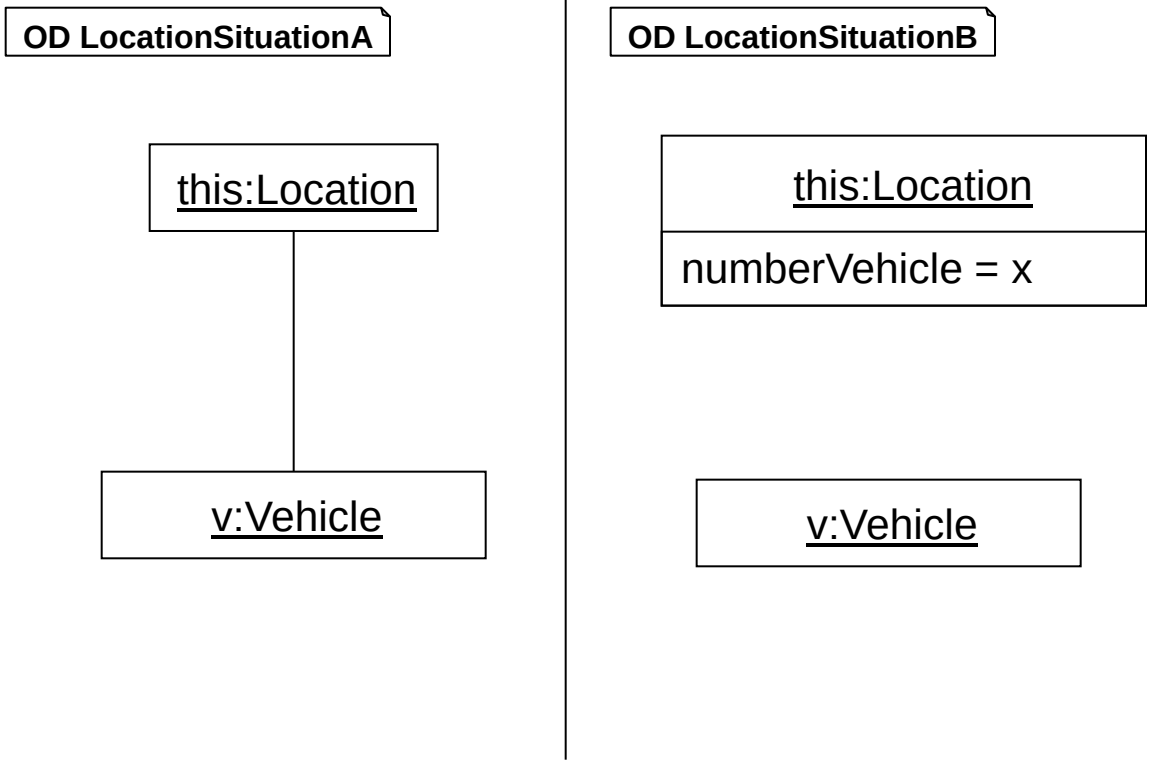
Task 6.2.1 – Pre- and Postconditions for Method Specifications by Object Diagrams

Change the following method specifications such that the pre- and postconditions are expressed by object diagrams and adjust the OCL expressions accordingly.

- When removing a vehicle it must belong to a location. After removing, the size of the list of the vehicles at the location is reduced by one.

```
context Location.removeVehicle(Vehicle v):
  pre:  vehicles.contains(v)
  post: !vehicles.contains(v) &&
        (vehicles@pre.size-1 == vehicles.size)
```

Solution 6.2.1




```
context Location.removeVehicle(Vehicle v)
  pre:  OD.LocationSituationA
  post: OD.LocationSituationB
        && (x == vehicles@pre.size-1)
```

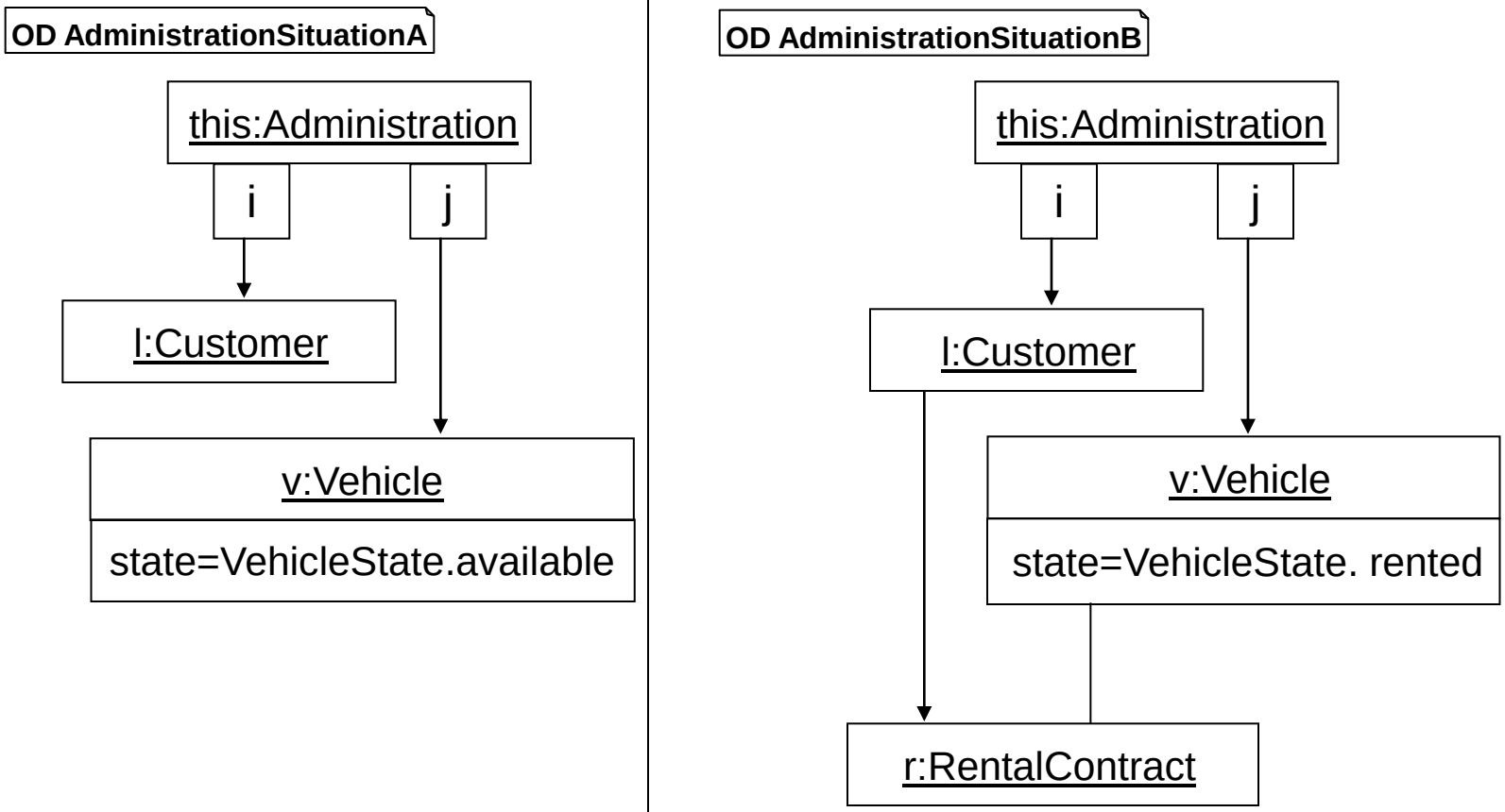
Task 6.2.2 – Pre- and Postconditions for Method Specifications by Object Diagrams

Change the following method specifications such that the pre- and postconditions are expressed by object diagrams and adjust the OCL expressions accordingly.

- During vehicle renting, the customer has to exist and the desired vehicle must be available. Customer and vehicle must be specified in the rental contract. The vehicle is afterwards in the state "rented". For this task, a simplified version of the rent()-method is used where only one vehicle can be rented.

```
context RentalContract Administration.rent(Customer l,
                                           Vehicle v):
  pre:  Administration.getAvailableVehicles().contains(v)
        && (exists c in Administration.customers: c == l)
  post: (result.getLeaser() == l)
        && (result.rentedCar == v)
        && (v.getState() == VehicleState.rented)
```

Solution 6.2.2

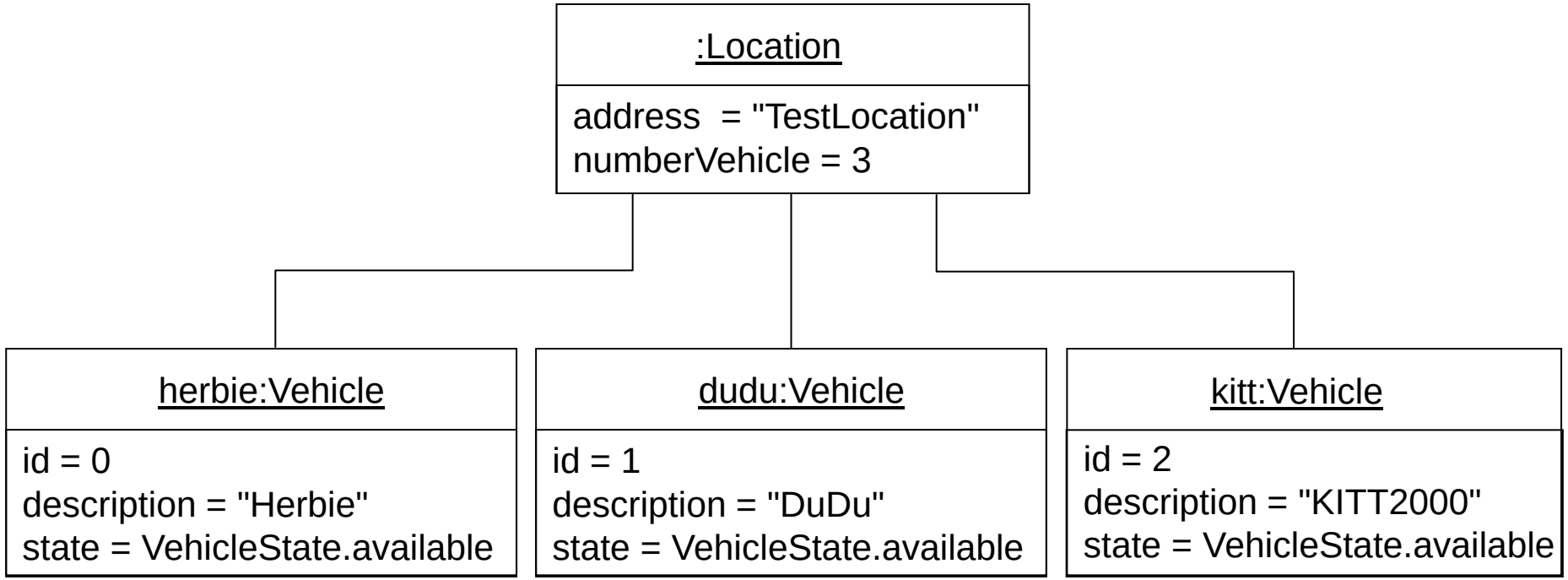



```
context RentalContract Administration.rent(Customer l,
                                           Vehicle v):
     pre:  OD.AdministrationSituationA
     post: OD.AdministrationSituationB &&
           (result == r)
```

Task 6.3 – Test Case Setup by Object Diagrams

```
public void testRemoveVehicle() {
  // Setup for the test case
  Location location = new Location("TestLocation", newTime(..));
  Vehicle herbie    = new Vehicle("Herbie", 0, VehicleState.available);
  Vehicle dudu      = new Vehicle("DuDu", 1, VehicleState.available);
  Vehicle kitt      = new Vehicle("KITT2000", 2, VehicleState.available);

  location.addVehicle(herbie);
  location.addVehicle(dudu);
  location.addVehicle(kitt);

  // The postcondition uses @pre, thus the list must be saved
  java.util.List oldVehicles = location.getVehicles().clone();

  // Examination of the precondition
  assertTrue(location.getVehicles().contains(herbie));

  location.removeVehicle(herbie);

  // Examination of the postcondition
  assertTrue(!location.getVehicles().contains(herbie) &&
             oldVehicles.size()-1 == location.getVehicles().size());
}
```

Solution 6.3


:Location
address = "TestLocation"
numberVehicle = 3
herbie:Vehicle
id = 0
description = "Herbie"
state = VehicleState.available
dudu:Vehicle
id = 1
description = "DuDu"
state = VehicleState.available
kitt:Vehicle
id = 2
description = "KITT2000"
state = VehicleState.available