SE SOFTWARE ENGINEERING

RWTH AACHEN

Exercise

Sequence Diagrams

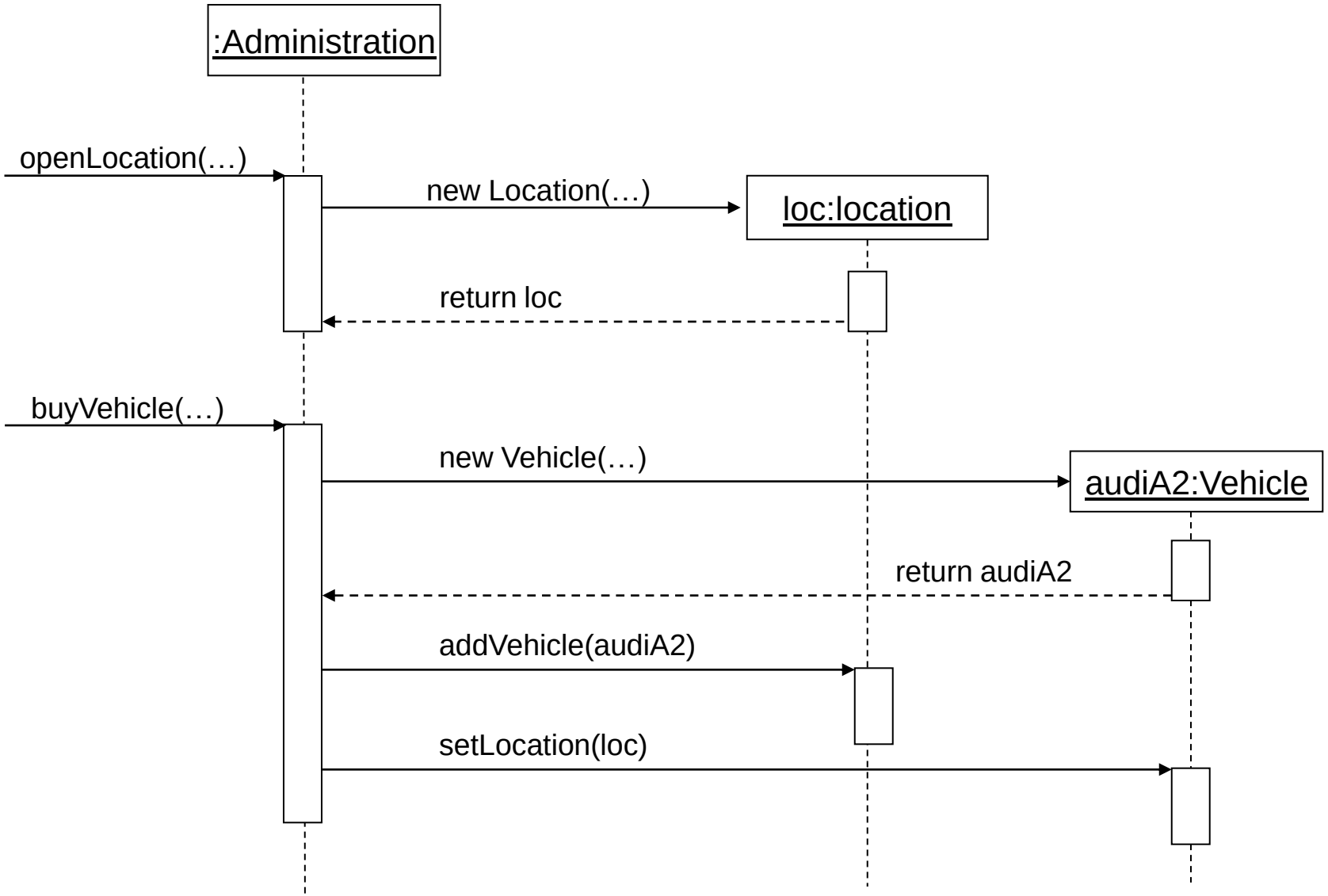
Prof. Dr. Bernhard Rumpe
Lehrstuhl für Software Engineering
RWTH Aachen

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

Exercise 8.1a – Modeling of Sequence Diagrams

Model sequence diagrams for the following workflows:

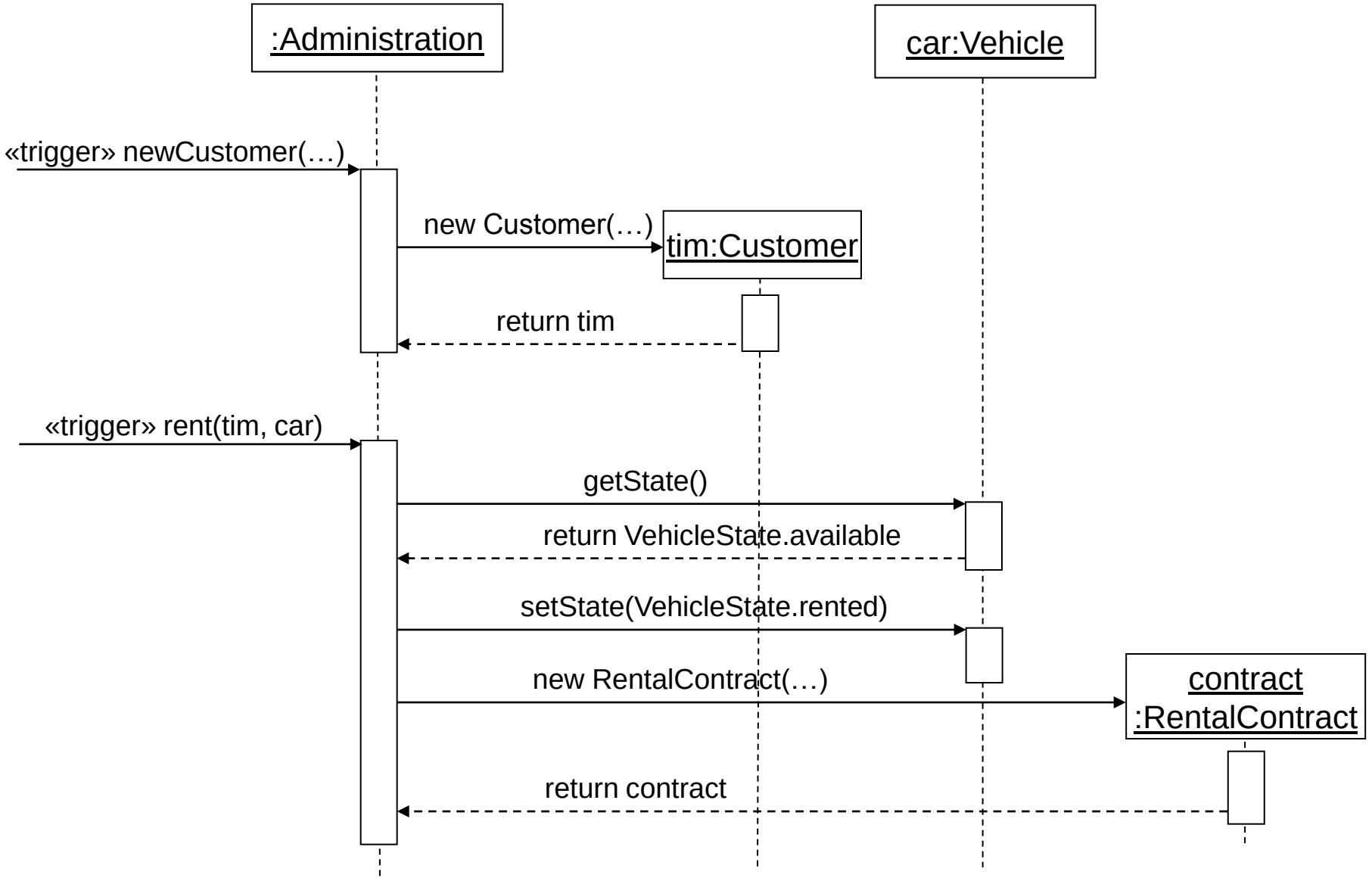
- Opening of a location with purchase of a vehicle "*Audi A2*" and an allocation to the new location.


:Administration
openLocation(…)
new Location(…)
loc:location
return loc
buyVehicle(…)
new Vehicle(…)
audiA2:Vehicle
return audiA2
addVehicle(audiA2)
setLocation(loc)

Exercise 8.1b – Modeling of Sequence Diagrams

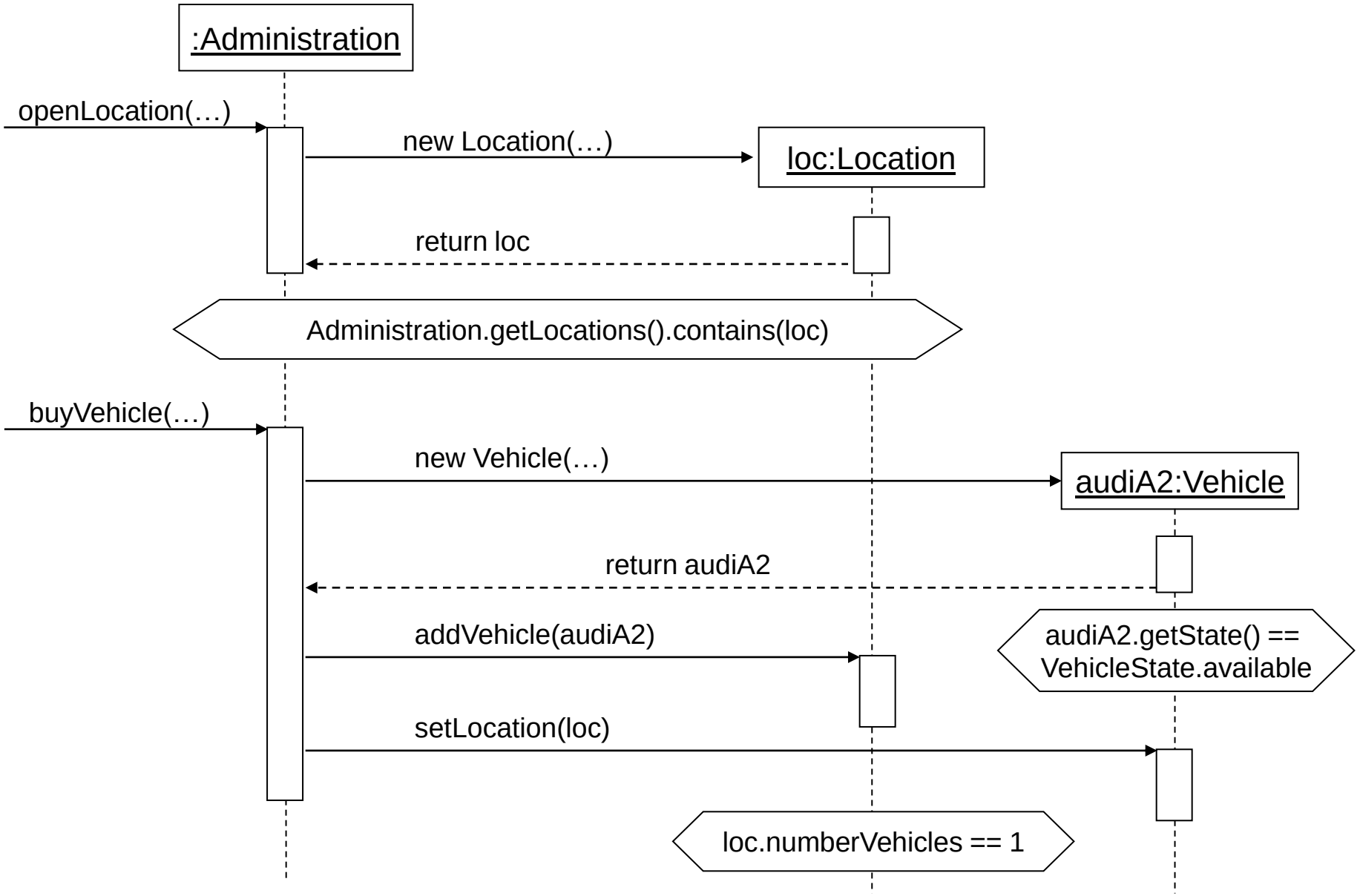
Model sequence diagrams for the following workflows:

- Creating a customer and renting of a vehicle.


:Administration
car:Vehicle
«trigger» newCustomer(…)
new Customer(…)
tim:Customer
return tim
«trigger» rent(tim, car)
getState()
return VehicleState.available
setState(VehicleState.rented)
new RentalContract(…)
contract
:RentalContract
return contract

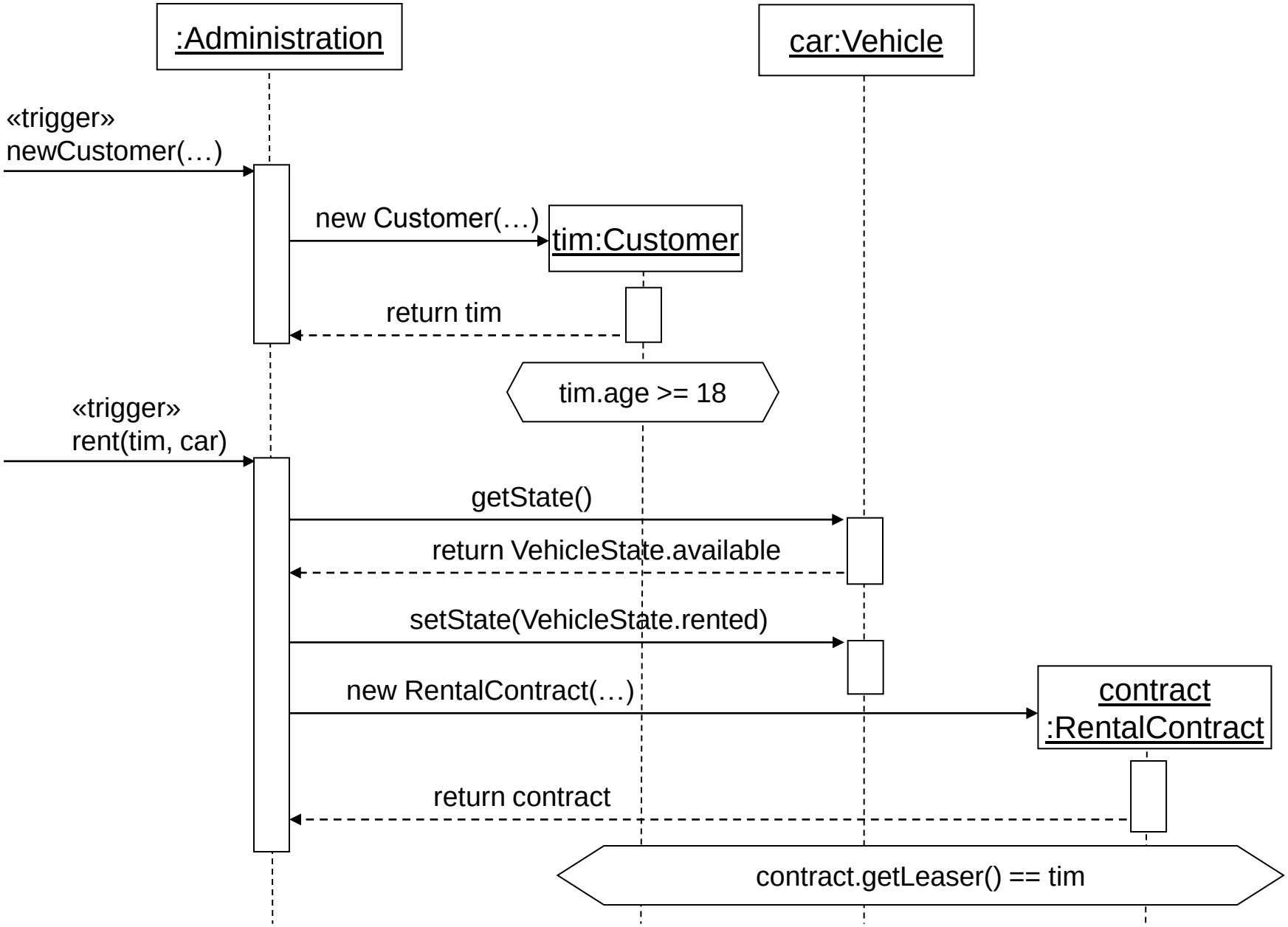
Exercise 8.2a – OCL Conditions in Sequence Diagrams

- Add now to the sequence diagrams from task 8.1a the following OCL conditions in a suitable place:
  - The number of vehicles at the new location is 1.
  - The "*Audi A2*" has the status available.
  - The new location is present in the location list of the administration.


:Administration
openLocation(…)
new Location(…)
loc:Location
return loc
Administration.getLocations().contains(loc)
buyVehicle(…)
new Vehicle(…)
audiA2:Vehicle
return audiA2
addVehicle(audiA2)
audiA2.getState() ==
VehicleState.available
setLocation(loc)
loc.numberVehicles == 1

Exercise 8.2b – OCL Conditions in Sequence Diagrams

- Add now to the sequence diagrams from task 8.2b the following OCL conditions in a suitable place:
  - The customer is at least 18 years old.
  - The customer is registered as a customer in the rental contract.


:Administration
car:Vehicle
«trigger»
newCustomer(…)
new Customer(…)
tim:Customer
return tim
tim.age >= 18
«trigger»
rent(tim, car)
getState()
return VehicleState.available
setState(VehicleState.rented)
new RentalContract(…)
contract
:RentalContract
return contract
contract.getLeaser() == tim

Exercise 8.3 – Test Case Generation from Sequence and Object Diagrams (1)

- The following JUnit test case is given:

```
public void testCase() {
  //Creation of objects
  Location aachen = new Location("Aachen");
  Location braunschweig = new Location("Braunschweig");
  Vehicle autoA1 = new Car("A1", 0, VehicleState.available);
  Vehicle autoBS1 = new Car("BS1", 1, VehicleState.available);

  //Linking objects
  aachen.addVehicle(autoA1);
  braunschweig.addVehicle(autoBS1);
  autoA1.setLocation(aachen);
  autoBS1.setLocation(braunschweig);

  //Perform test
  Location loc = autoA1.getLocation();
  loc.removeVehicle(autoA1);
  braunschweig.addVehicle(autoA1);
  autoA1.setLocation(braunschweig);

  ...
```

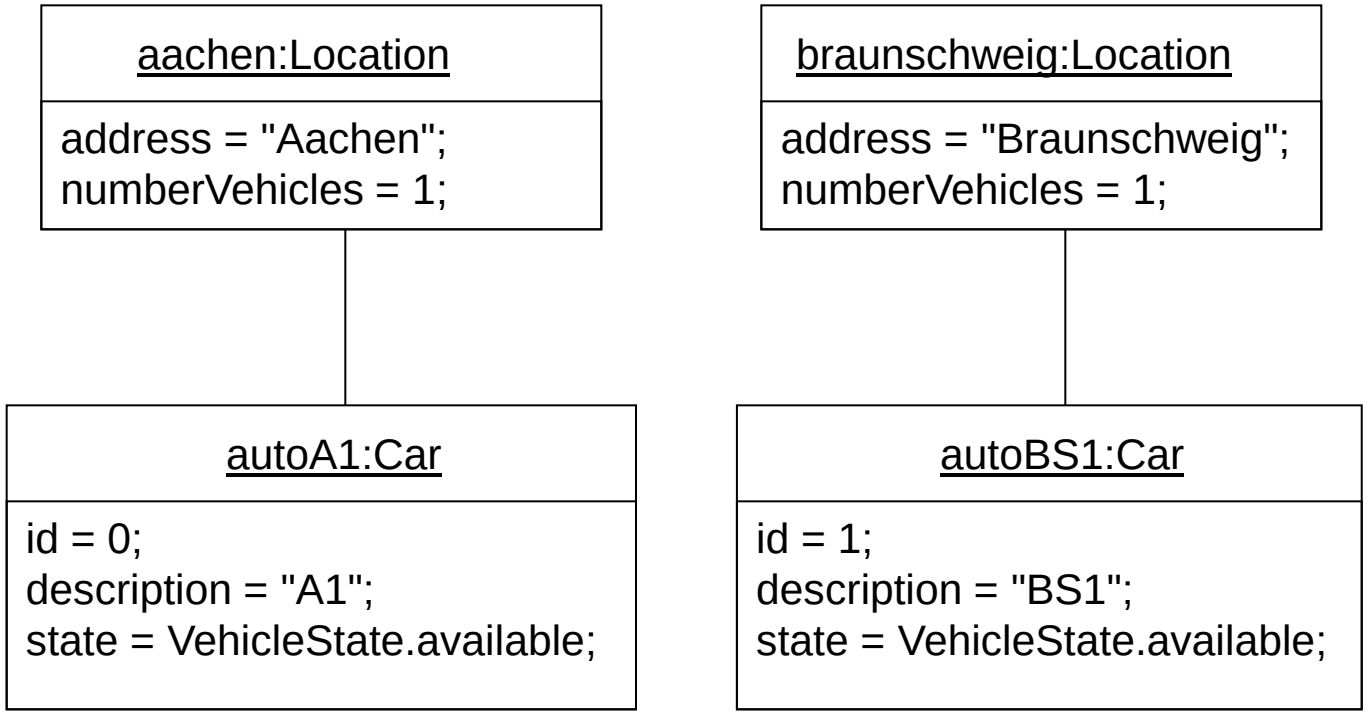
Exercise 8.3 – Test Case Generation from Sequence and Object Diagrams (2)

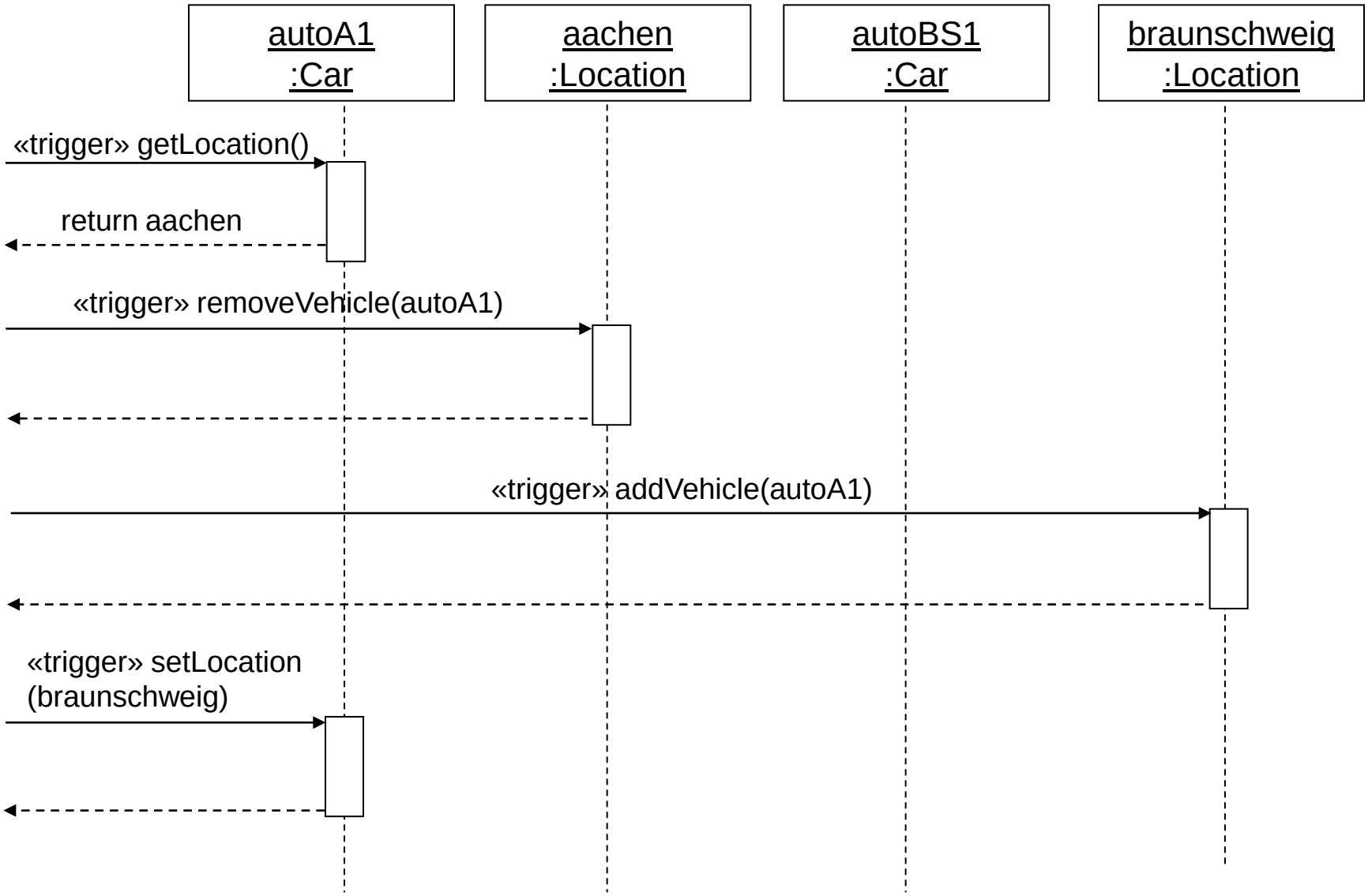
```
  …

  //Examination of the results
  assertEquals(aachen.getVehicles().size(), 0);
  assertEquals(aachen.getAddress(), "Aachen");

  assertEquals(braunschweig.getVehicles().size(), 2);
  assertEquals(braunschweig.getAddress(), "Braunschweig");

  assertTrue(braunschweig.getVehicles().contains(autoA1));
  assertTrue(braunschweig.getVehicles().contains(autoBS1));
  assertFalse(aachen.getVehicles().contains(autoA1));

  assertEquals(autoA1.getLocation(), braunschweig);
  assertEquals(autoBS1.getLocation(), braunschweig);
}
```

Exercise 8.3 – Test Case Generation from Sequence and Object Diagrams (3)

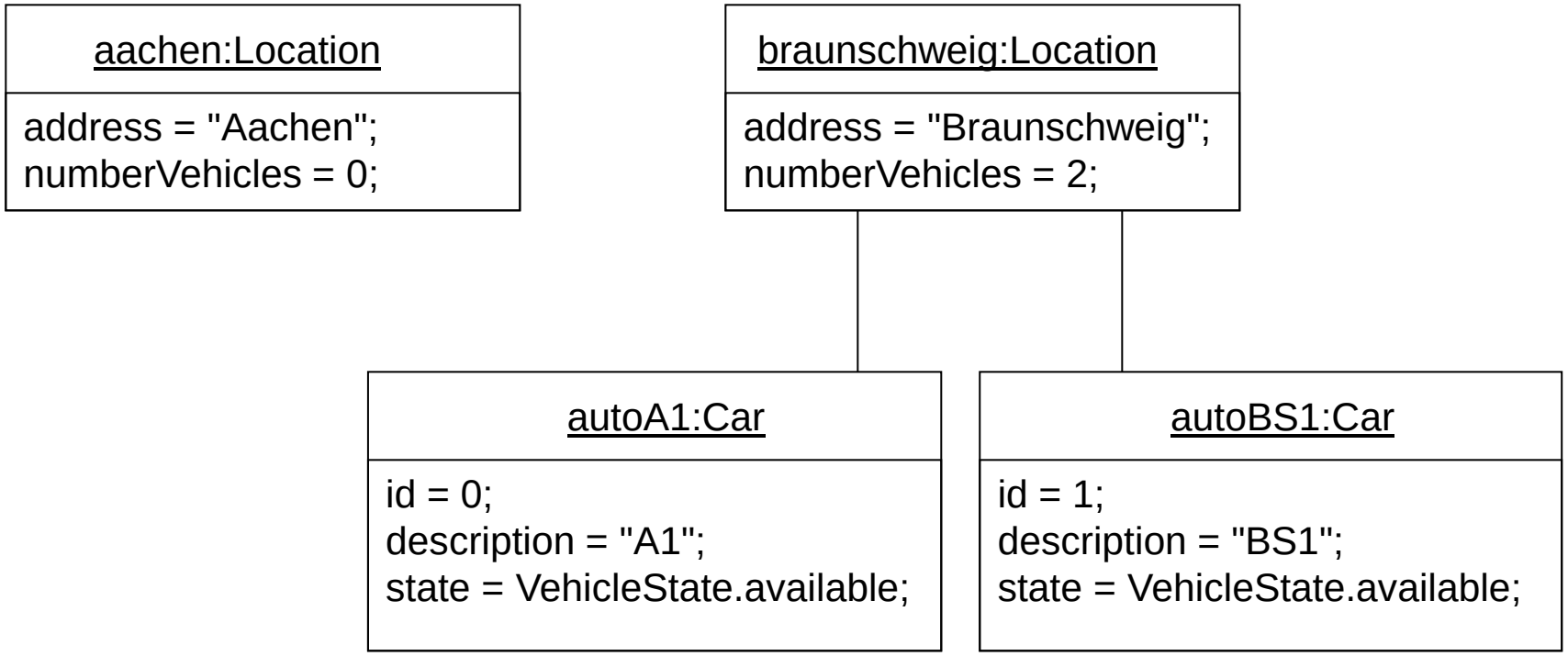
- Extract from the source code the following diagrams:
  - An object diagram, which represents the structure of the case of test
  - A sequence diagram which describes the expiration of the case of test
  - An object diagram for the examination of the test drop result

OD: Structure – Object Diagram




autoA1
:Car
aachen
:Location
autoBS1
:Car
braunschweig
:Location
«trigger» getLocation()
return aachen
«trigger» removeVehicle(autoA1)
«trigger» addVehicle(autoA1)
«trigger» setLocation
(braunschweig)

OD: Examination – Object Diagram



Codegeneration from Sequence Diagrams

- The previous examples just used the triggers from the sequence diagrams
- But: How to check the other communications?
- Ideas?

Code-Instrumentation

Generated/Java

```
public class LocationInstr
extends Location {
  …
  public void addVehicle(Vehicle v) {
    Object caller = Logger.getCurrentCaller();
    Object callee = this;
    Logger.log(caller, callee, "addVehicle", v);
    Logger.addCurrentCaller(this);
    super.addVehicle(v);
    Logger.log(caller, callee, "addVehicle", v);
    Logger.removeCurrentCaller(this);
  }
  …
}
```

- Method- and attribute-calls are logged using code-instrumentation
- Subclass encapsulates instrumentation
- Factory enables to start instrumentation dynamically (next slide)

Codeinstrumentation using Factories