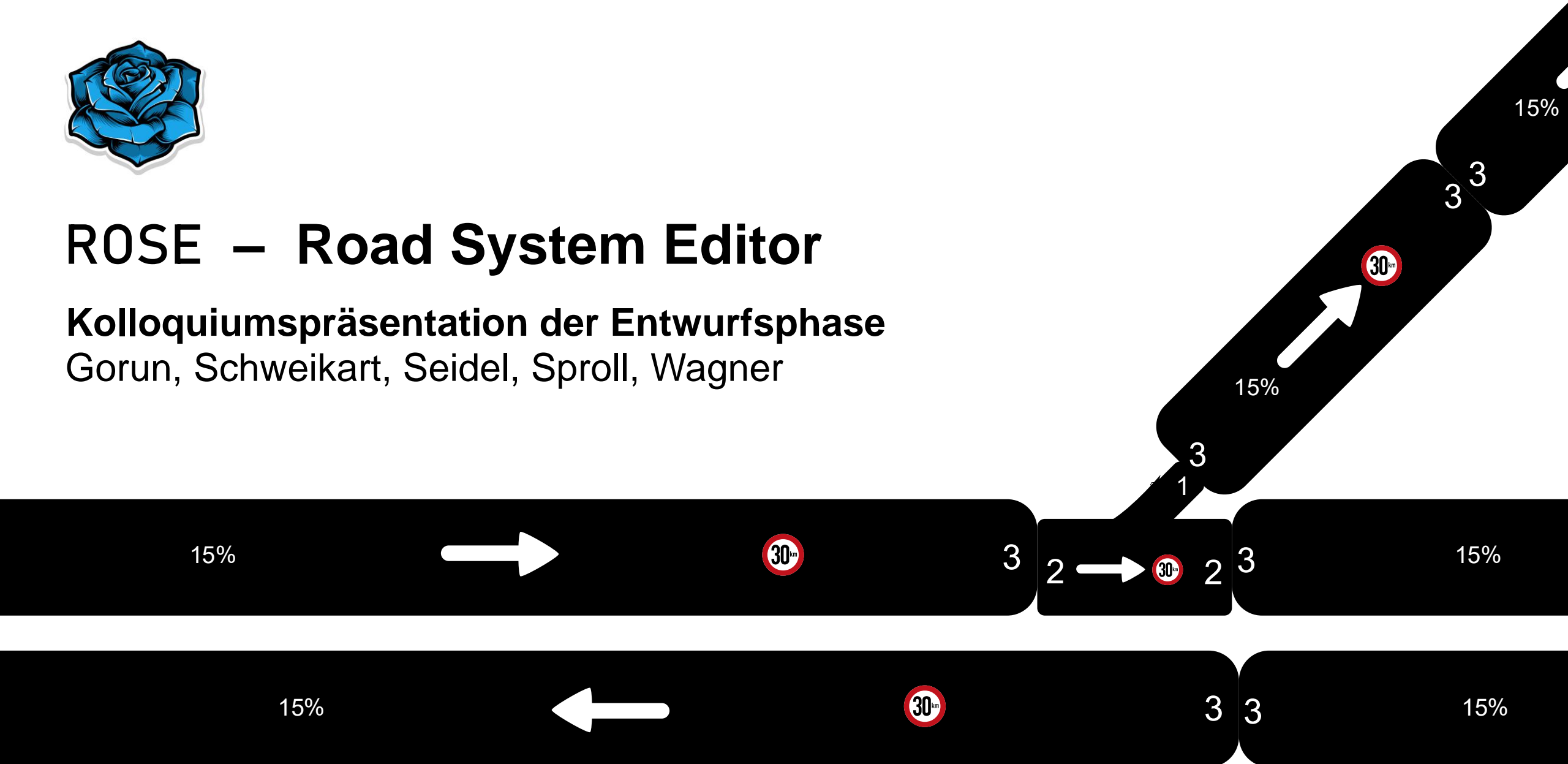


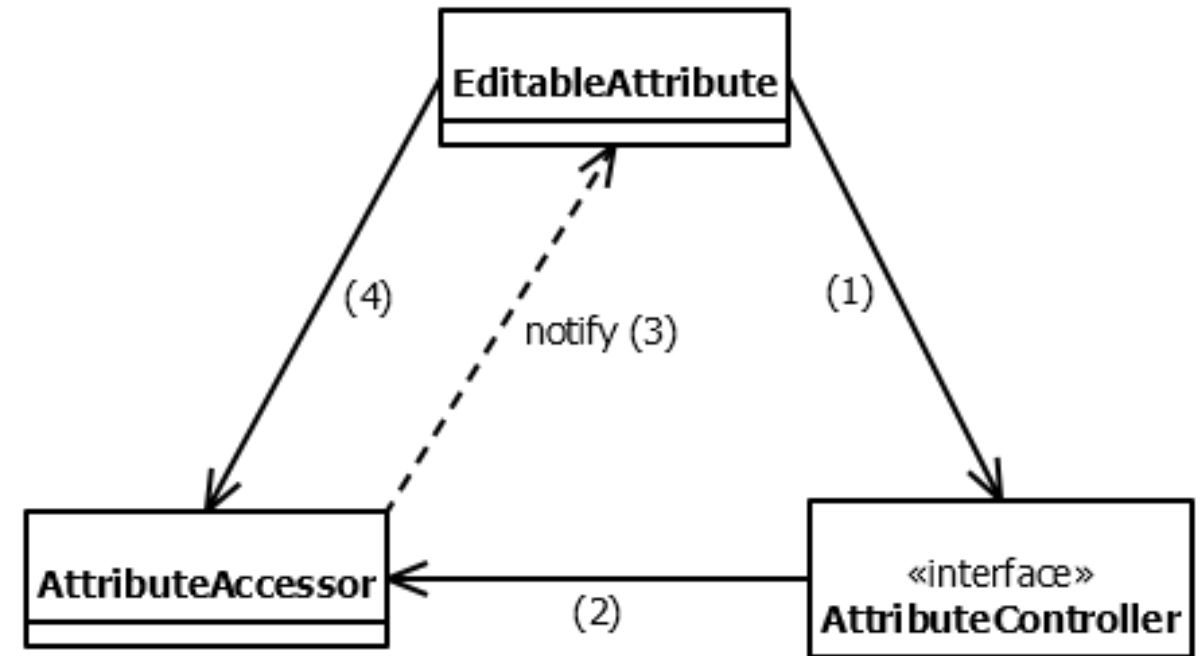


ROSE – Road System Editor

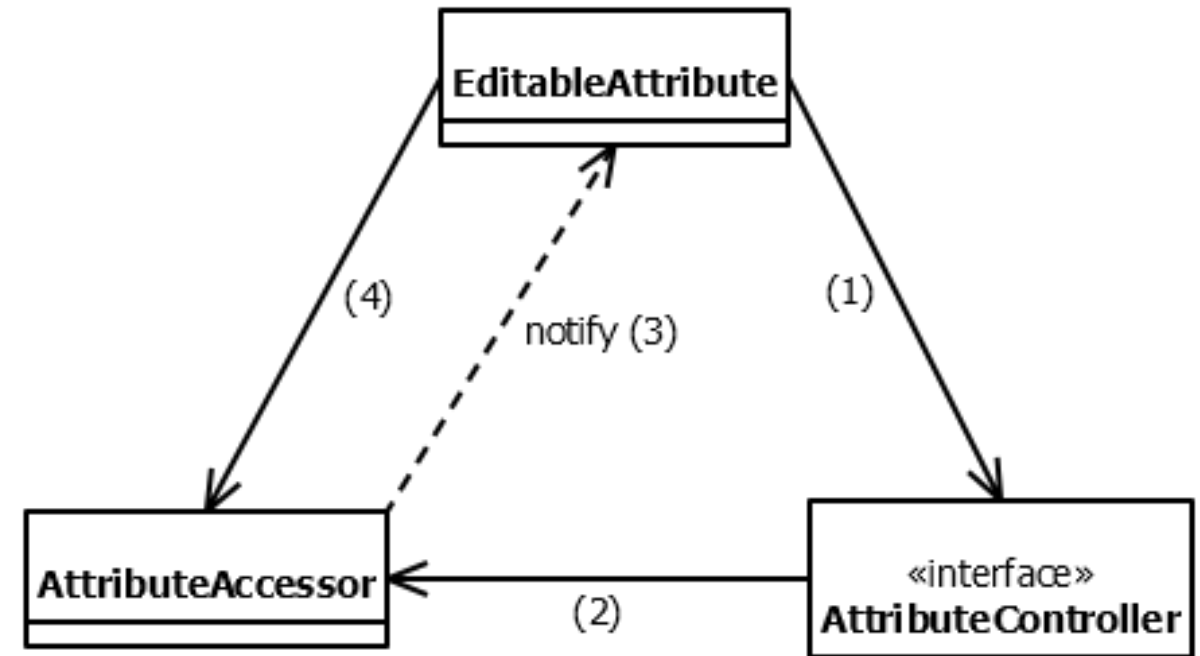
Kolloquiumspräsentation der Entwurfsphase
Gorun, Schweikart, Seidel, Sproll, Wagner



■ Architekturmuster: Model-View-Controller



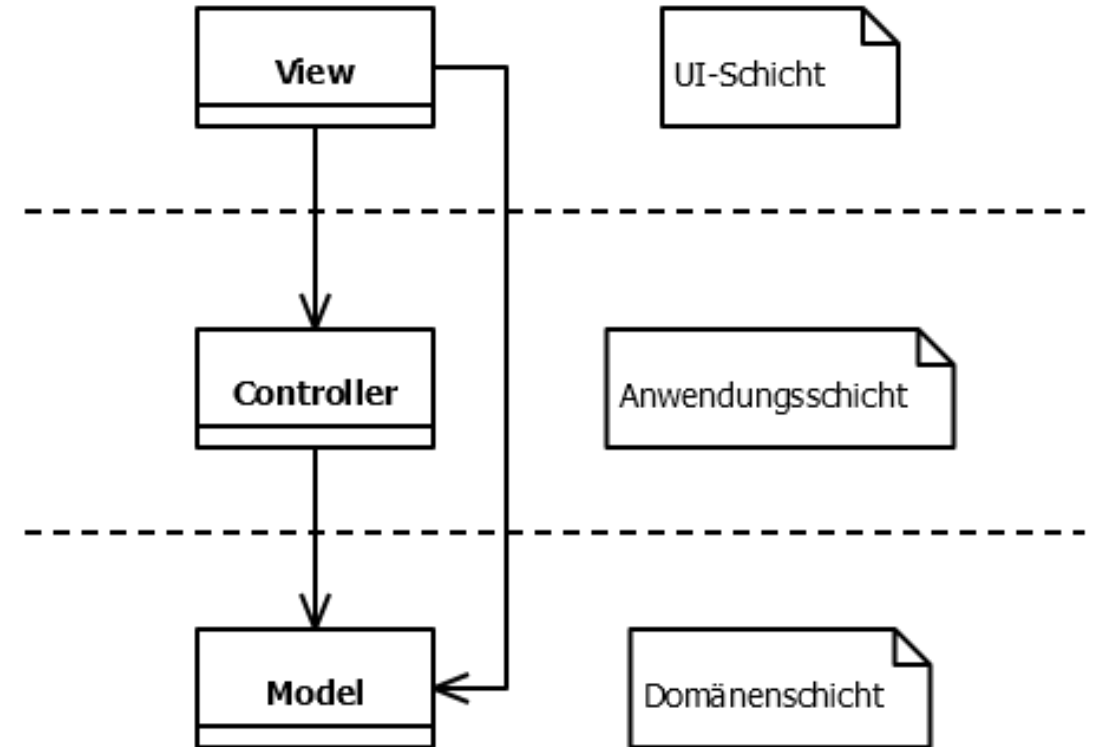
- Architekturmuster:
Model-View-Controller
- Observer-Pattern



Architektur



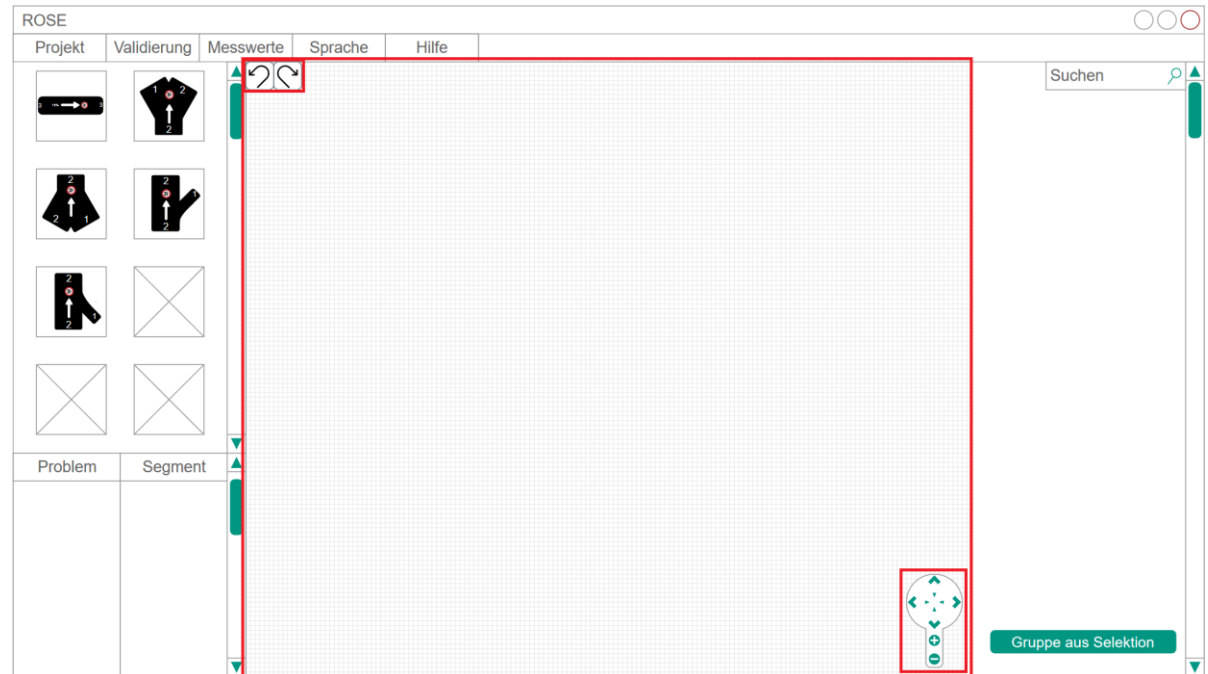
- Architekturmuster:
Model-View-Controller
- Observer-Pattern
- Transparente Schichtenarchitektur



Architektur der View



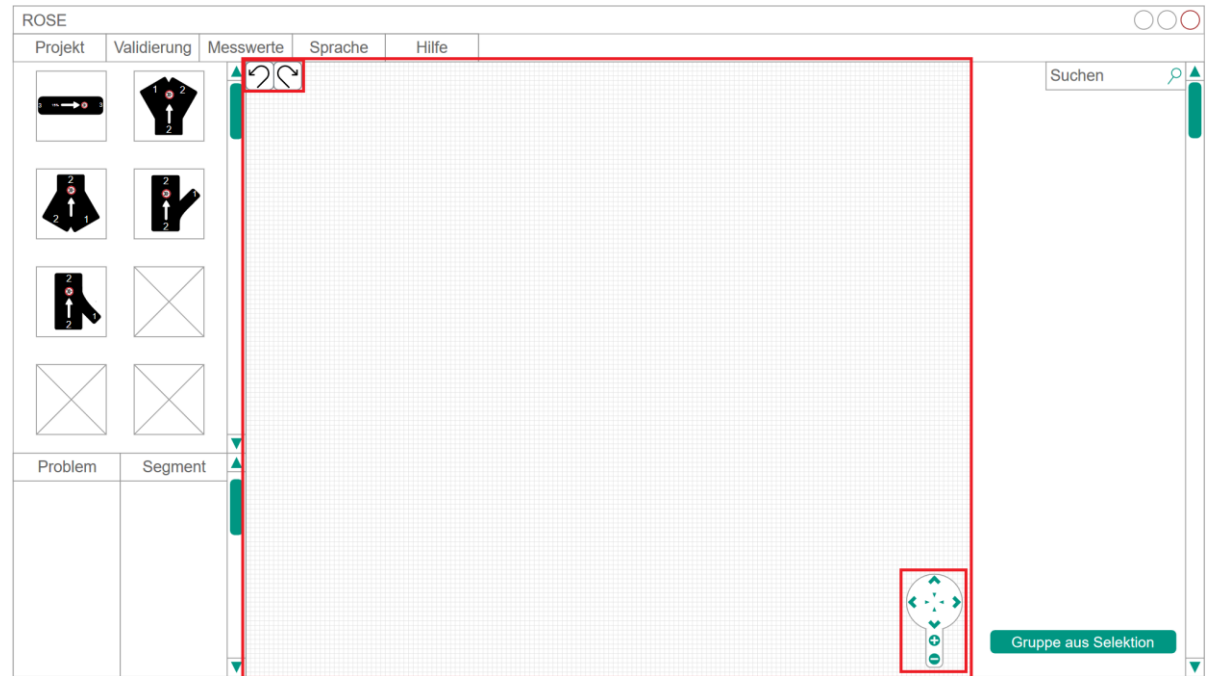
- Besteht aus Windows und Panels



Architektur der View

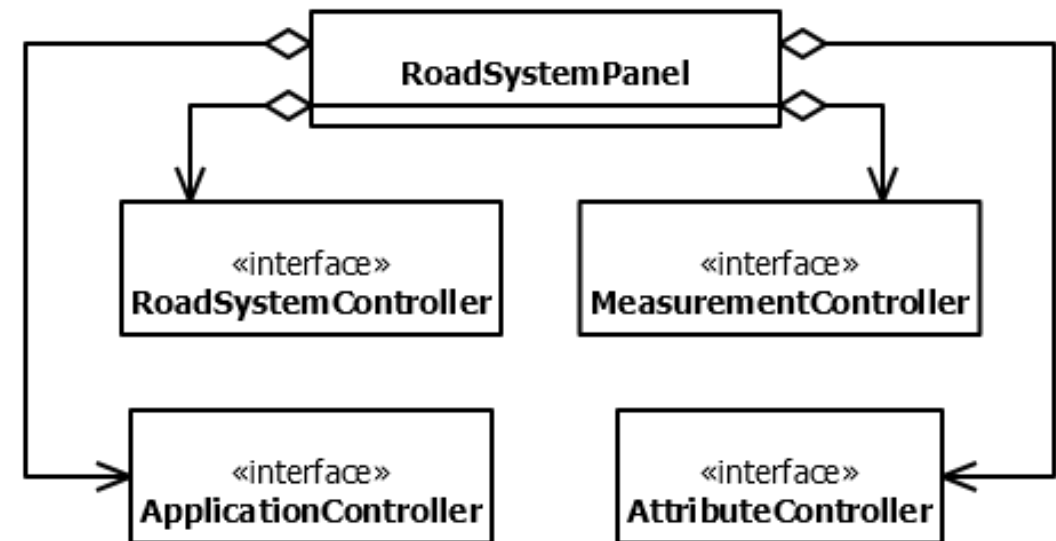


- Besteht aus Windows und Panels
- Kompositum



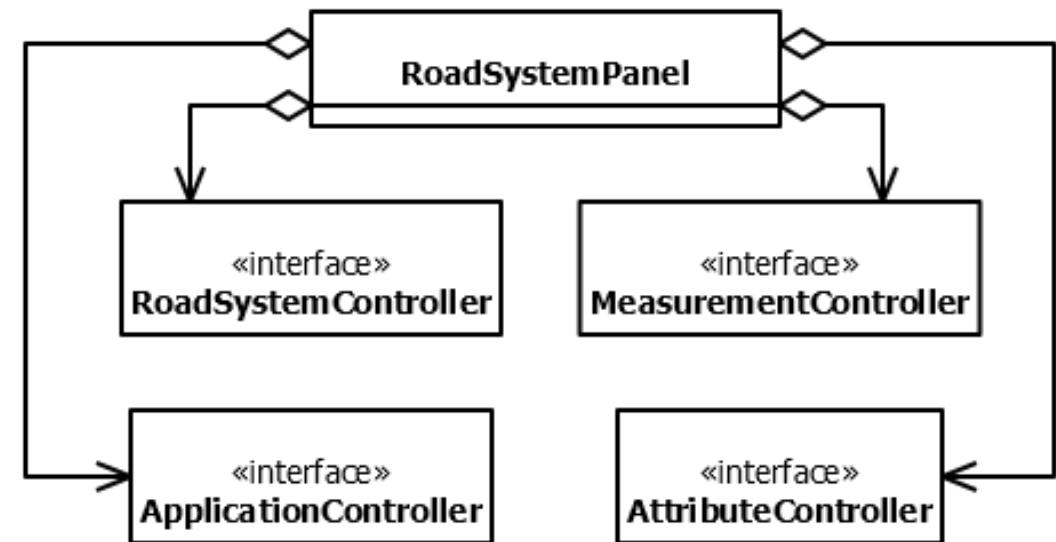
Architektur der Controller

- Aufteilung nach Funktionsbereichen

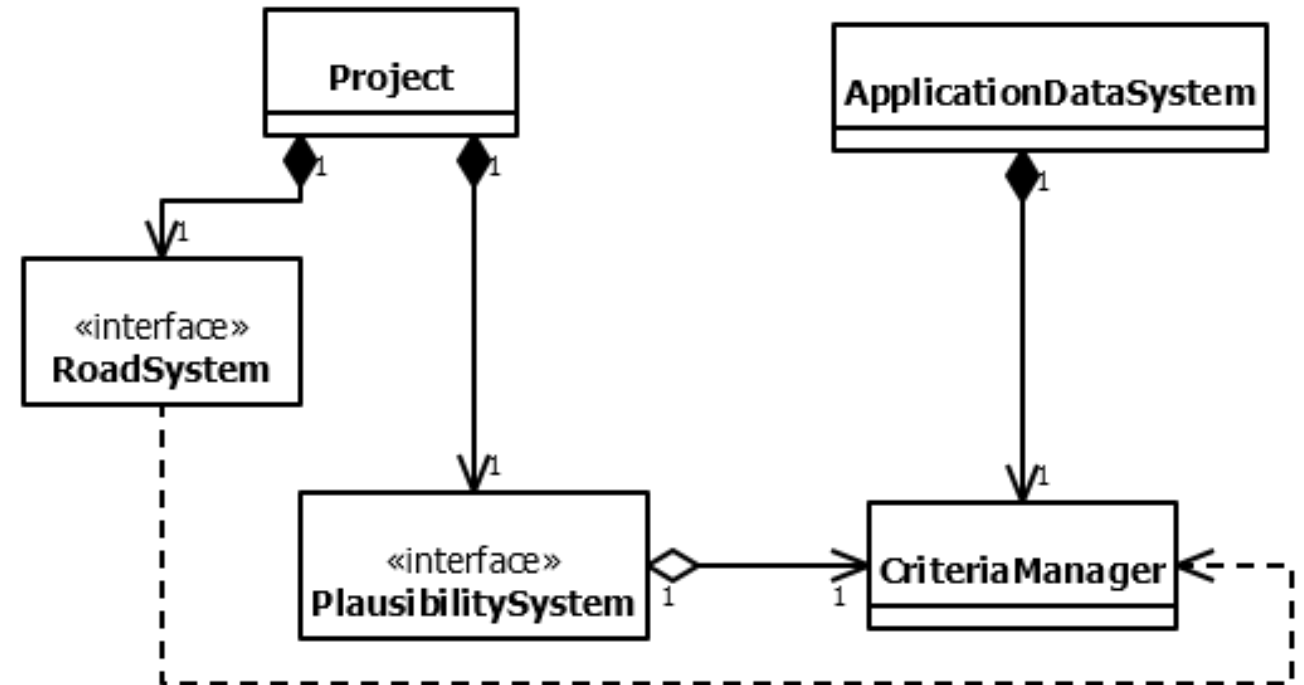


Architektur der Controller

- Aufteilung nach Funktionsbereichen
- Koordination von Operationen auf des Models



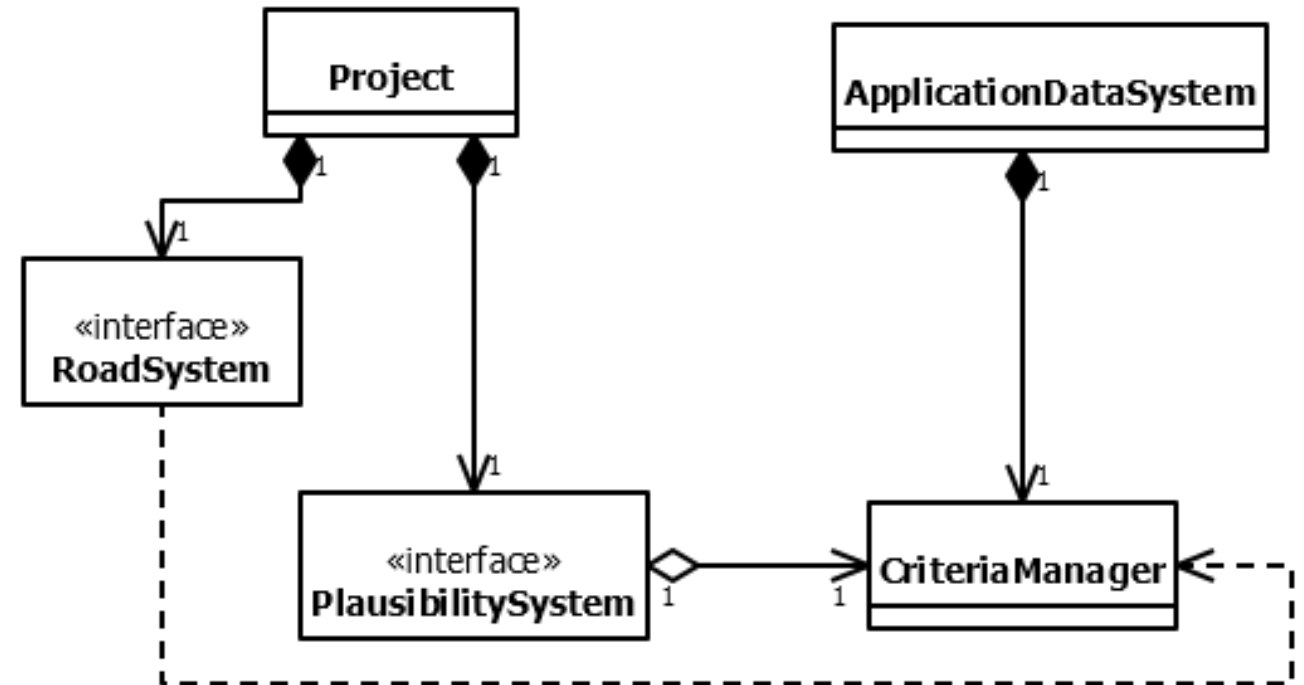
■ Aufteilung in Subsysteme



Architektur des Models

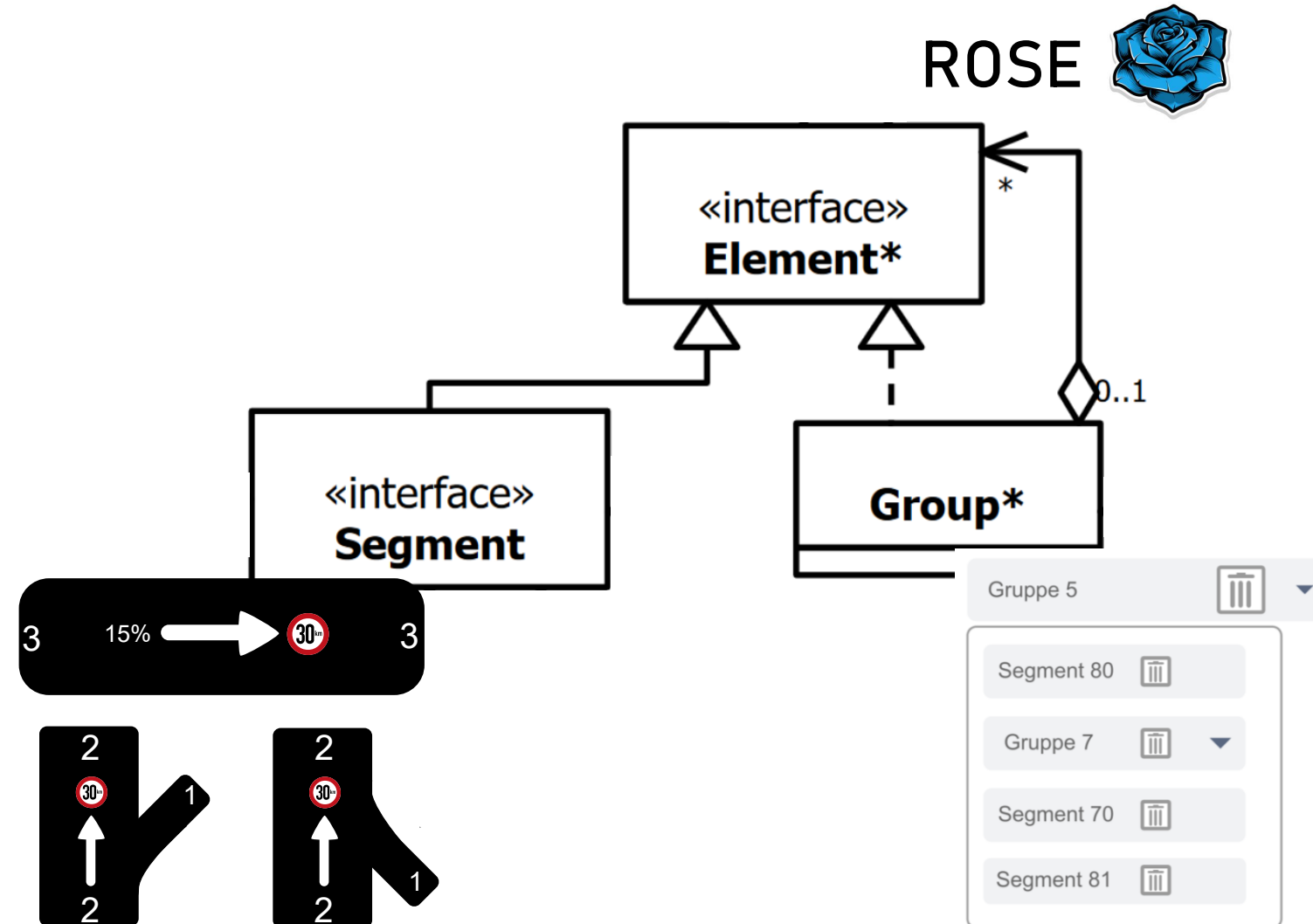


- Aufteilung in Subsysteme
- Fassade als Schnittstelle



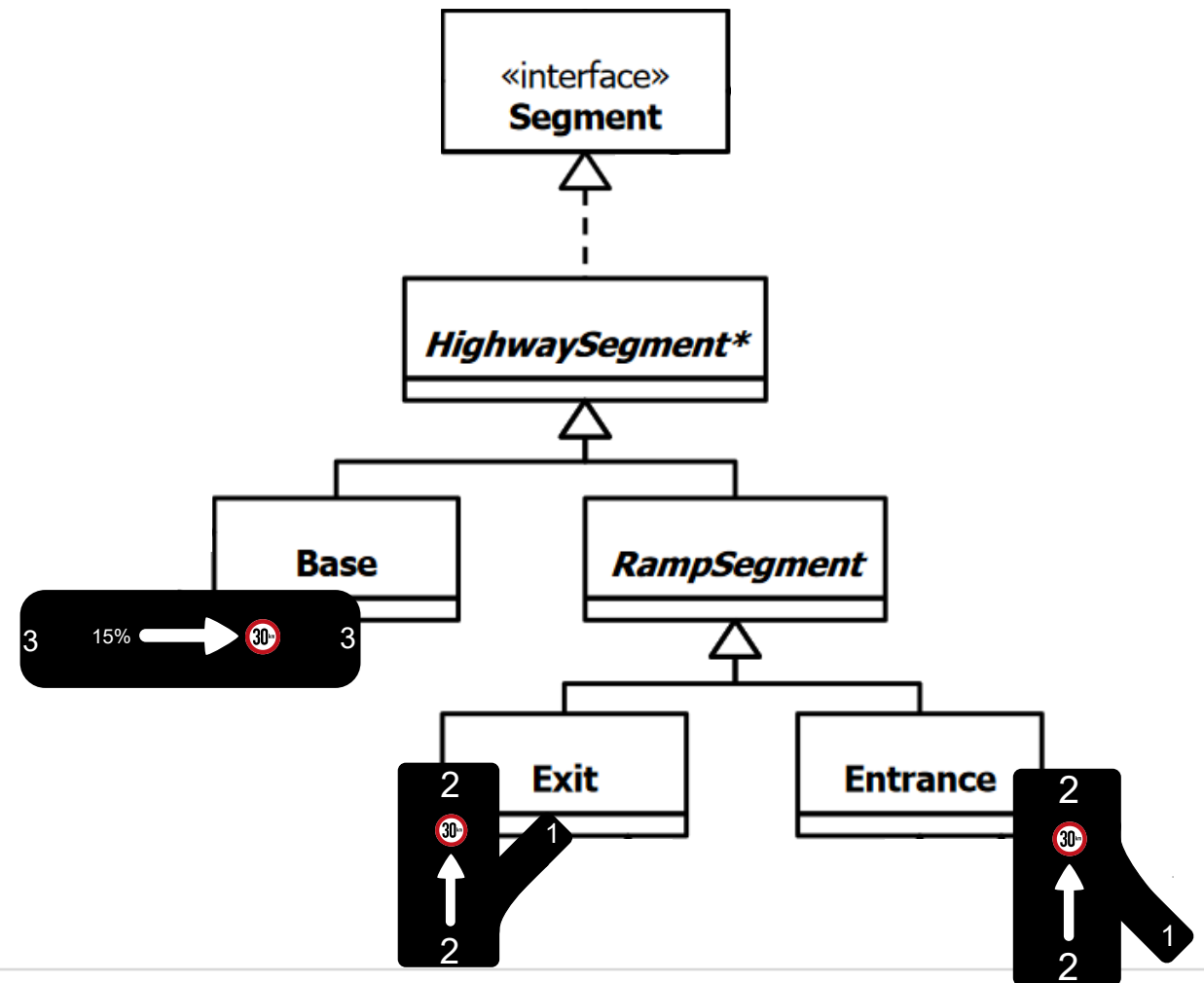
Straßennetz

■ Elemente als Kompositum

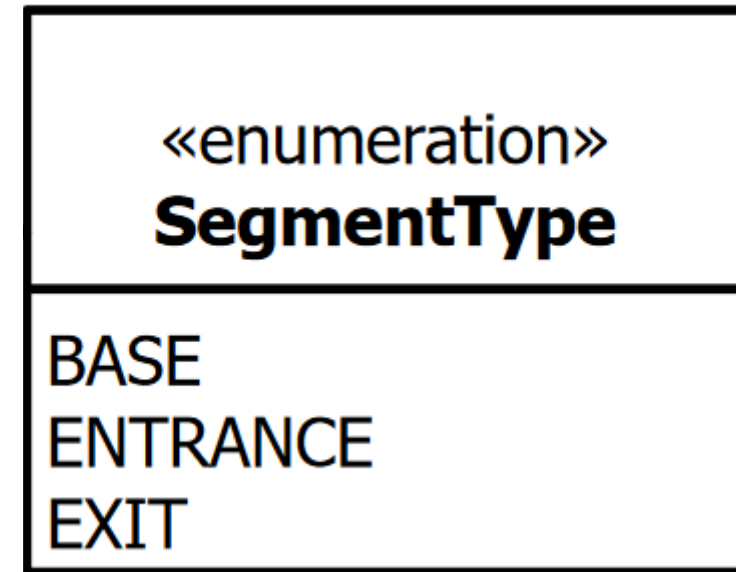


Straßennetz

- Elemente als Kompositum
- Segmentarten durch Vererbung

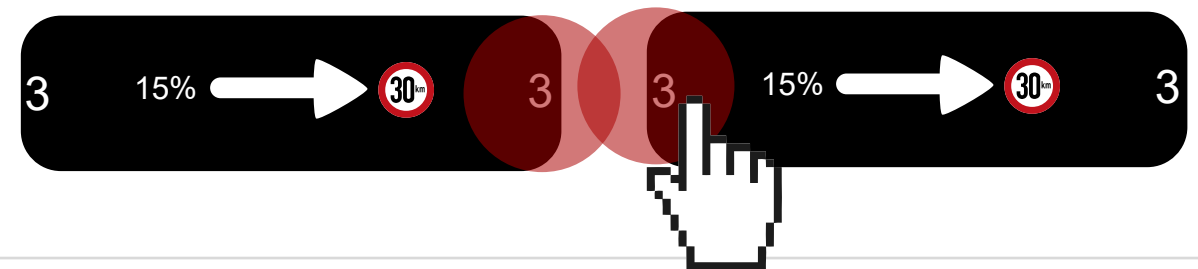
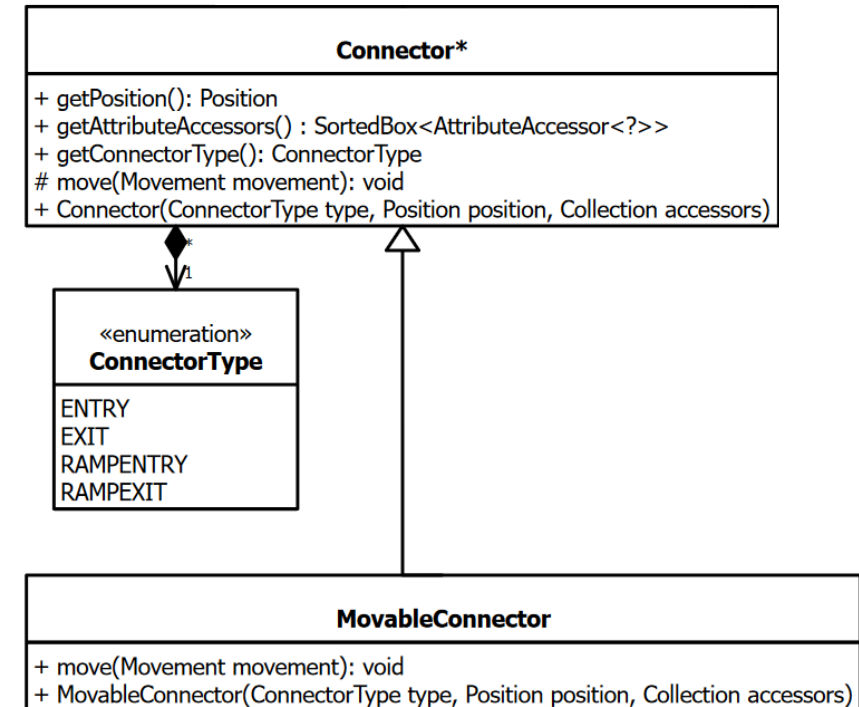


- Elemente als Kompositum
- Segmentarten durch Vererbung
- Identifikation durch Segmenttypen



Straßennetz

- Elemente als Kompositum
- Segmentarten durch Vererbung
- Identifikation durch Segmenttypen
- Endpunkte durch Connectoren

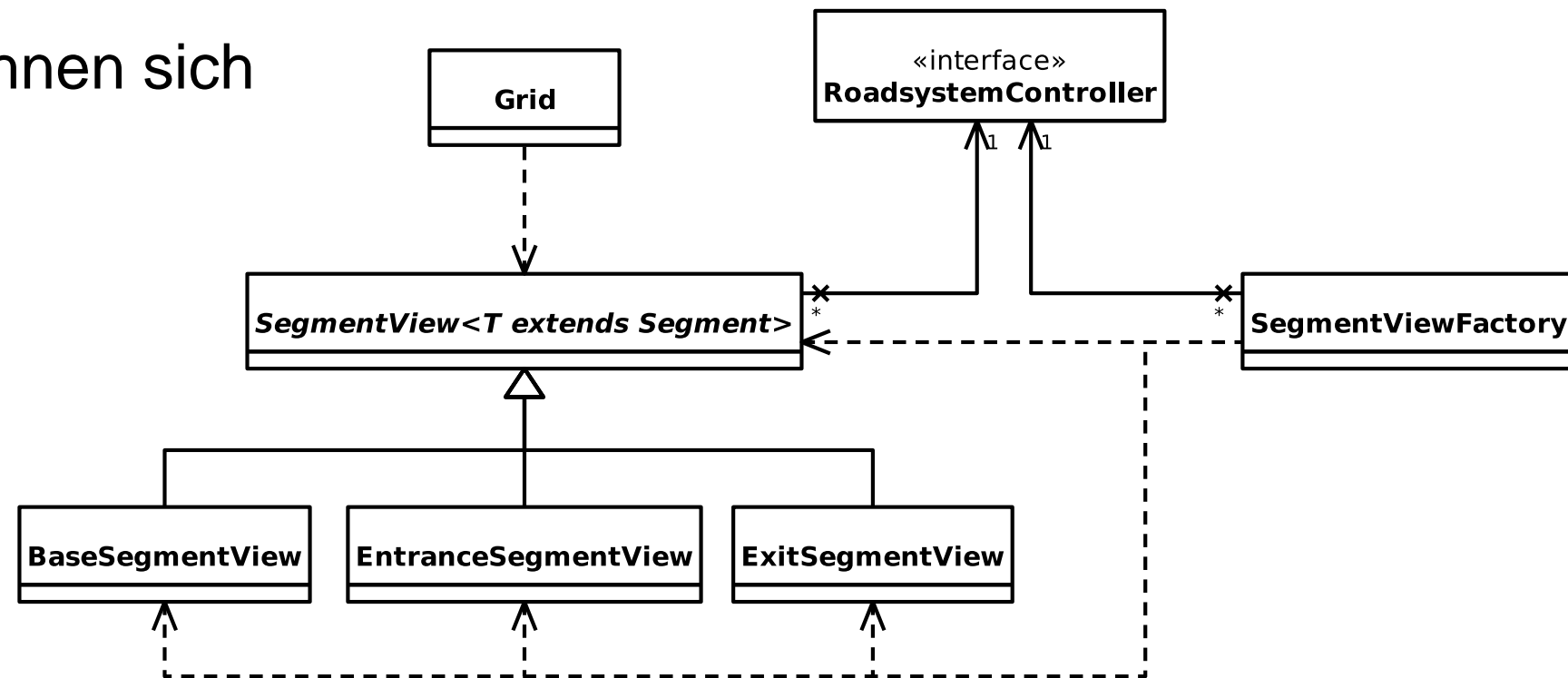


- Elemente als Kompositum
- Segmentarten durch Vererbung
- Identifikation durch Segmenttypen
- Endpunkte durch Connectoren
- Verwaltung durch Roadsystem Schnittstelle

```
«interface»
RoadSystem*
+ getElements() : Box<Element>
+ getElementsByName(String name) : Box<Element>
+ createSegment(SegmentType segmentType): void
+ createGroup(Collection includedElements): void
+ removeElement(Element element): void
+ getSharedAttributeAccessors(Collection elements) : SortedBox<AttributeAccessor<?>>
+ connectConnectors(Connector segment1Connector, Connector segment2Connector): void
+ disconnectConnection(Connection connection): void
+ disconnectFromAll(Segment segment): void
+ getAdjacentSegments(Segment segment) : Box<Segment>
+ getRootElements() : Box<Element>
+ getConnection(Connector connector): Connection
+ getConnections(Segment segment) : Box<Connection>
+ getConnections(Segment segment1, Segment segment2) : Box<Connection>
+ moveSegments(Collection segments): void
+ rotateSegment(Segment segment): void
+ rotateSegments(Collection segments): void
+ getTimeSliceSetting(): TimeSliceSetting
```

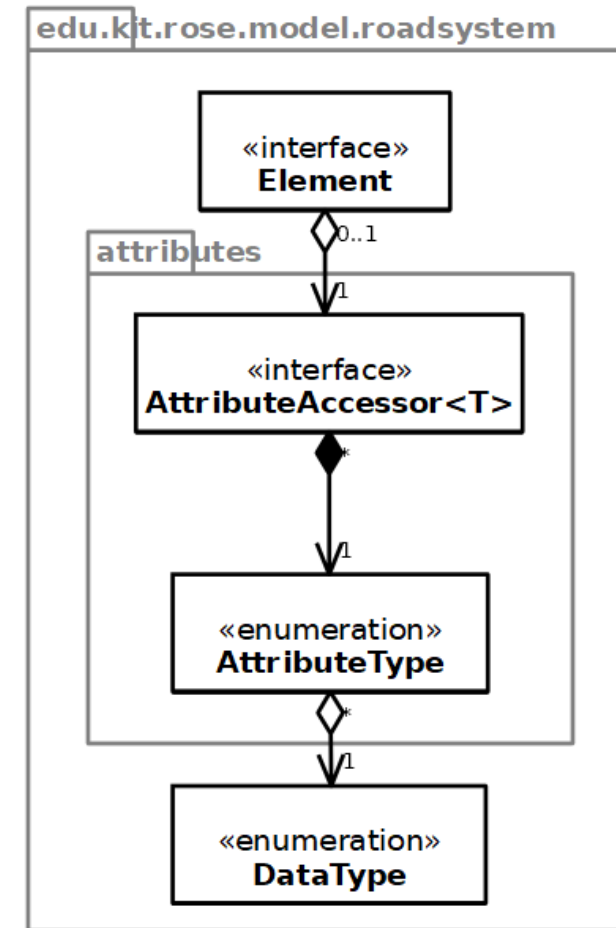
- Gespiegelte Struktur in der View

- SegmentViews zeichnen sich selbst



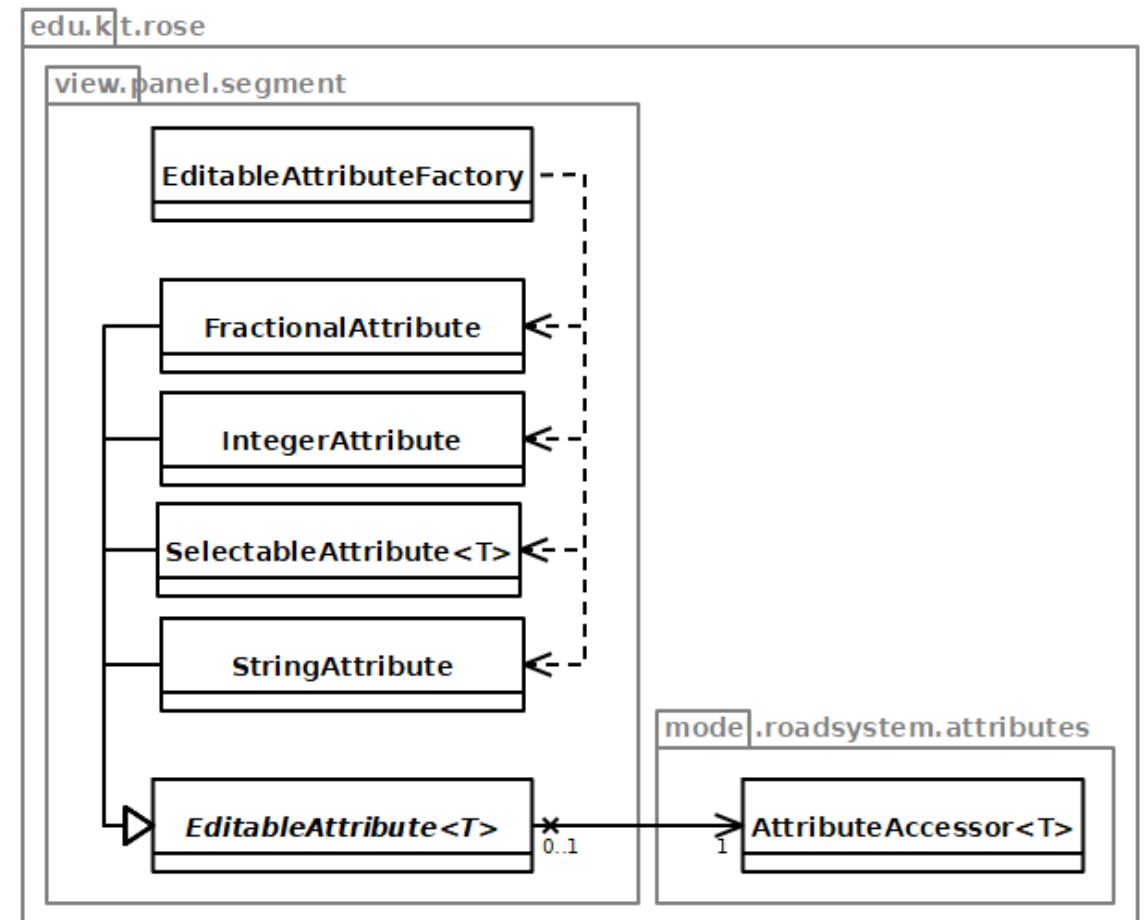
Attribute

- Segments stellen Liste von AttributeAccessors bereit



Attribute

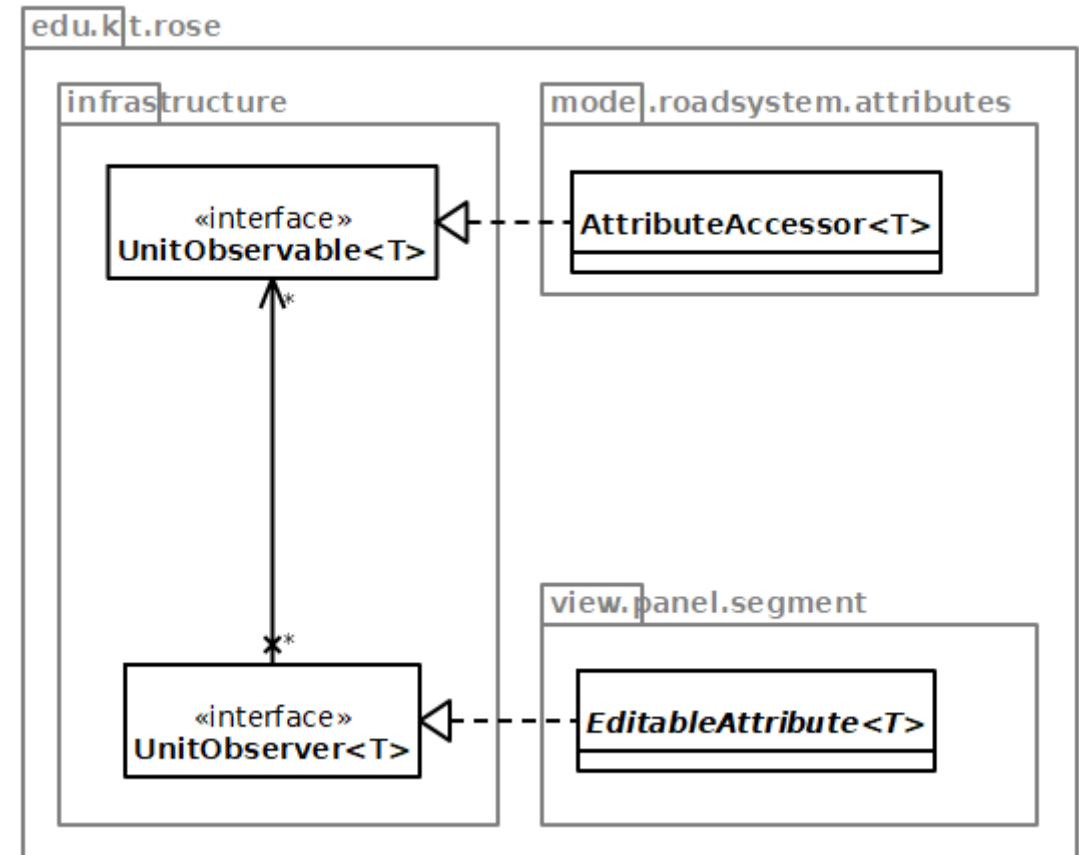
- Segments stellen Liste von AttributeAccessors bereit
- View stellt AttributeAccessor je nach DataType dar



Attribute



- Segments stellen Liste von AttributeAccessors bereit
- View stellt AttributeAccessor je nach DataType dar
- Kommunikation über UnitObserver und AttributeController

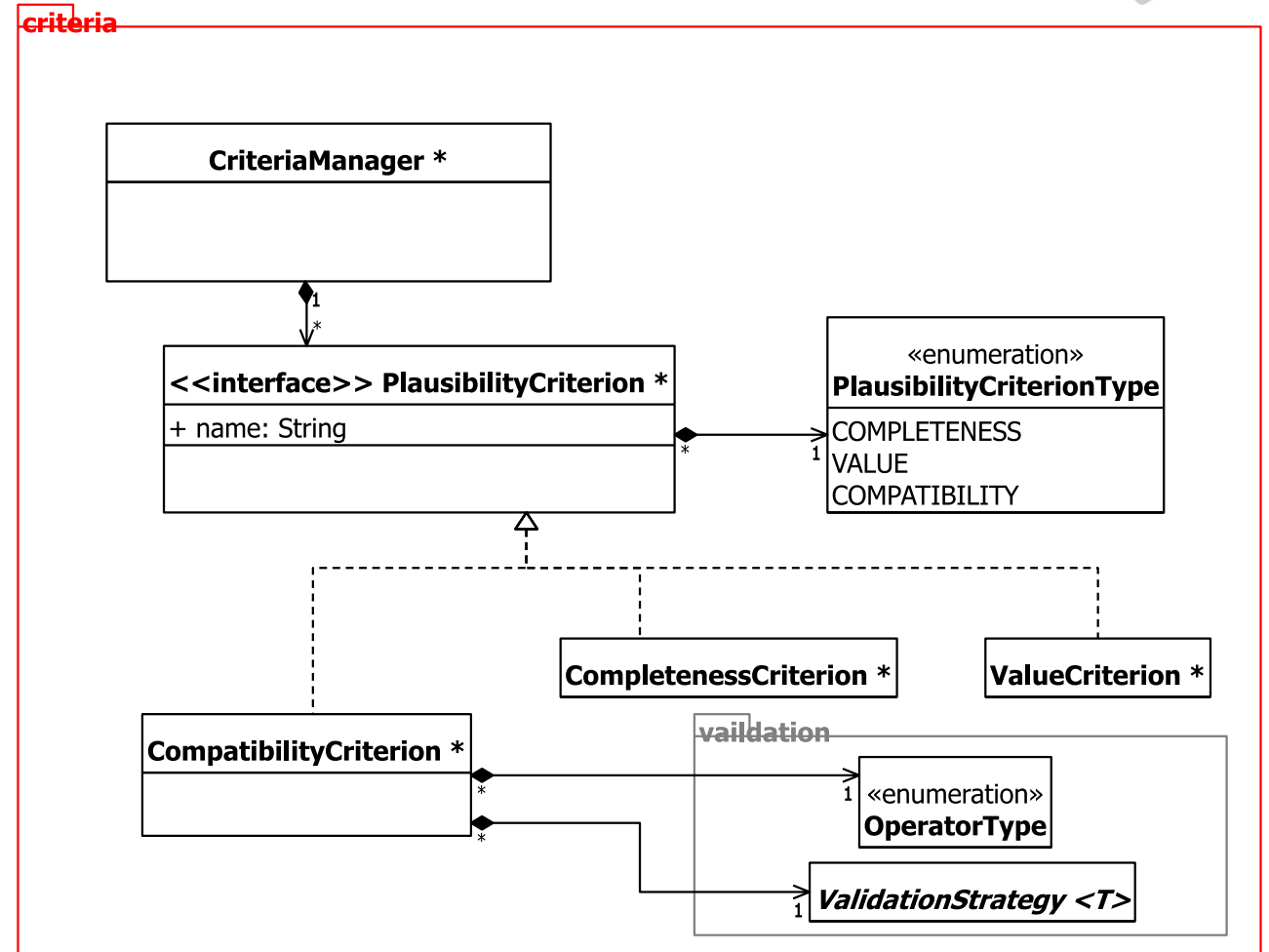


Plausibilitätskriterien



- Plausibilitätskriterien werden als globale Daten behandelt

- Import / Export



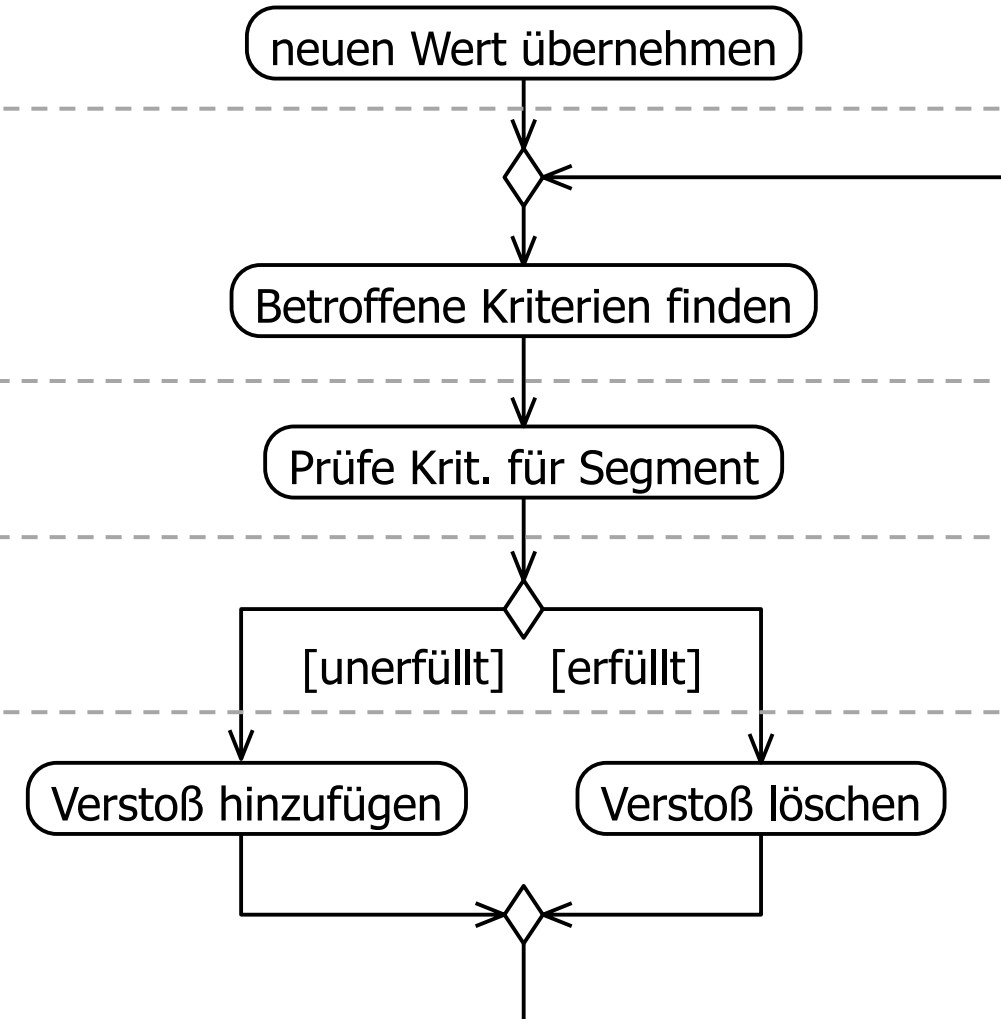
Plausibilitätskriterien

■ Segment

■ CriteriaManager

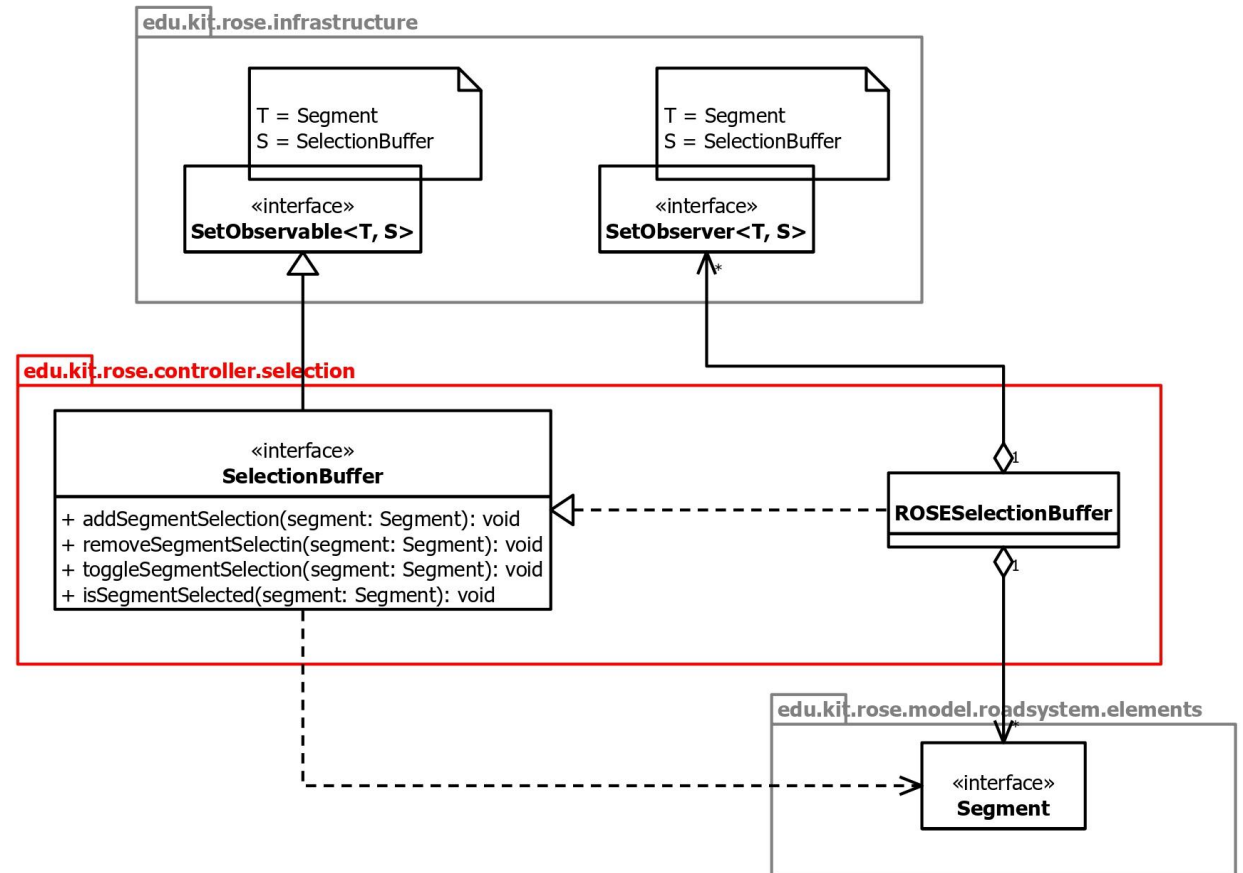
■ PlausibilityCriterion

■ ViolationManager

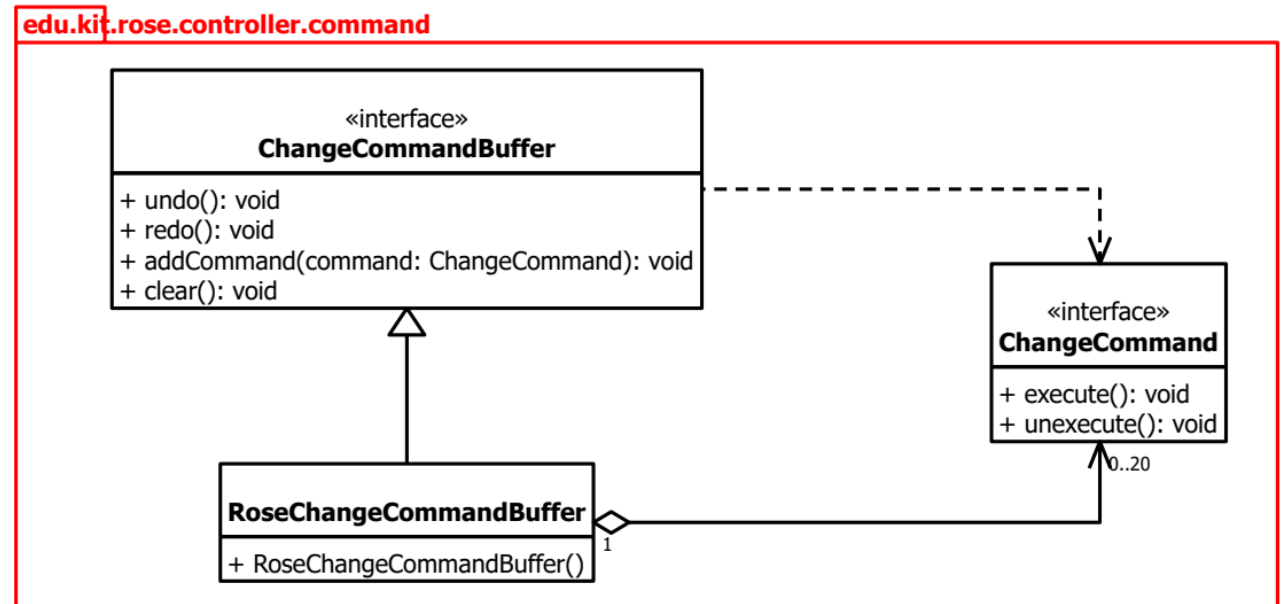


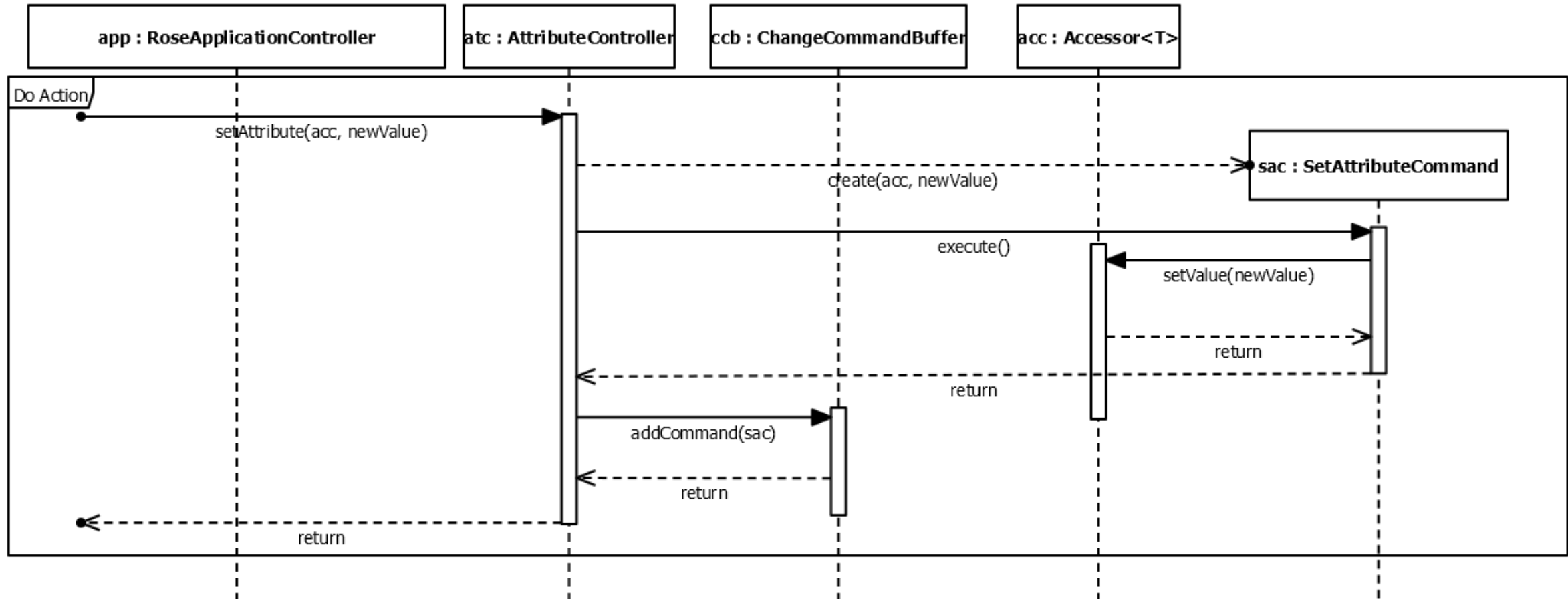
Workflow

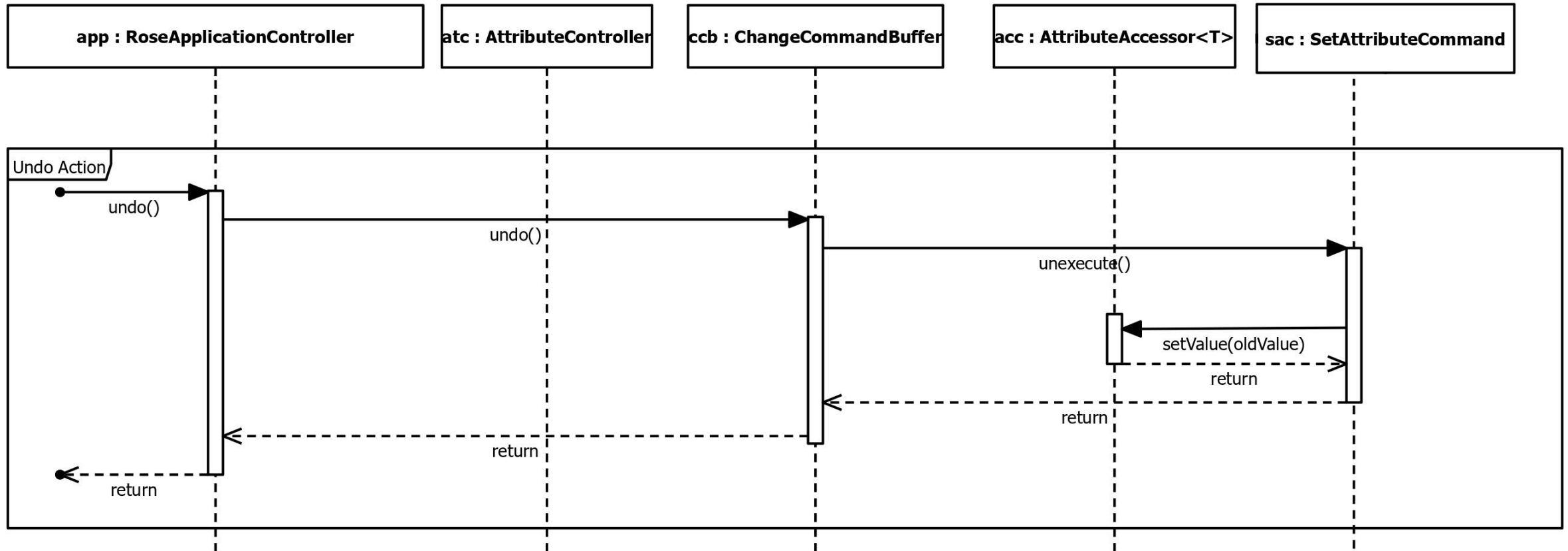
- Die selektierten Segmente werden in SelectionBuffer gespeichert



- Undo / Redo durch Command-Pattern
- Controller teilen sich eine ChangeCommandBuffer-Instanz







Änderungen



- Tutorial nicht Teil des Entwurfs

Änderungen

- Tutorial nicht Teil des Entwurfs
- Kleine Änderungen an den Mockups des Pflichtenhefts

- Logo von inksyndromeartwork, bereitgestellt über [www.freepik.com](https://de.freepik.com/inksyndromeartwork).
<https://de.freepik.com/inksyndromeartwork>