PG CHILL3

(Cooperation of Humans, Infrastructure and Linked Limousines)

Review III

16.04.2021

Projektgruppe CHILL 3.0

- Vier Studierende
  - Master Wirtschafts-/Informatik

- Lehrveranstaltung Projektgruppe
  - CHILL – Coorperation of Humans, Infrastructure and Linked Limousines
  - 10 Monate (01.06.2020 bis 16.04.2021)

- In Kooperation mit
  - OFFIS e.V. – Institut für Informatik
  - IAV GmbH Ingenieurgesellschaft Auto und Verkehr

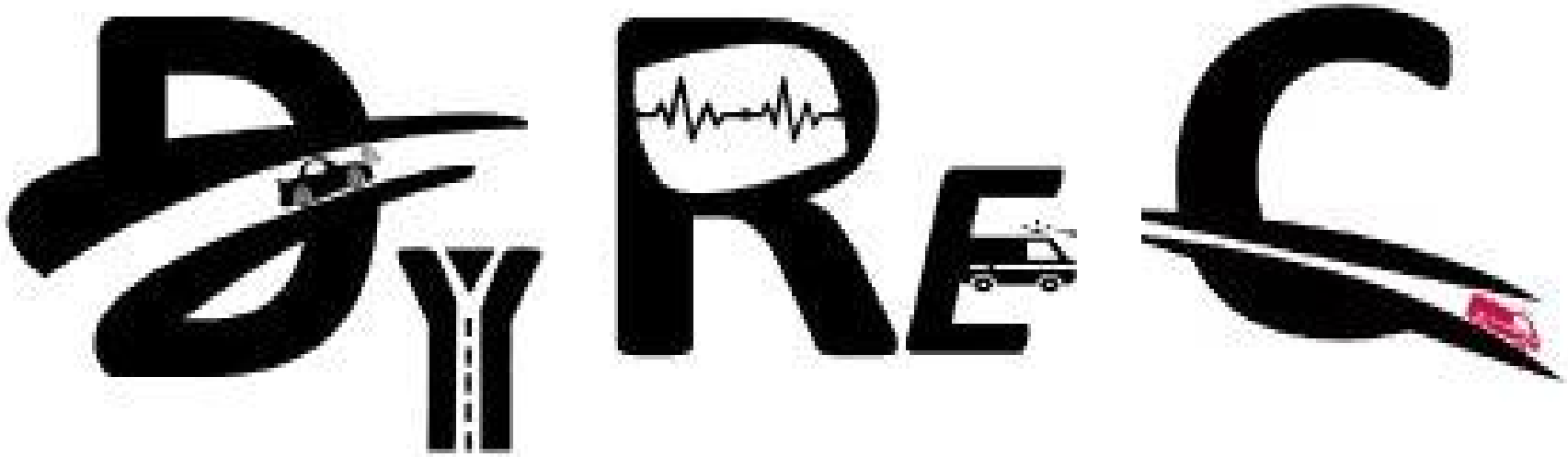
Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Demonstrator
- Evaluation
- Ausblick
- Fazit

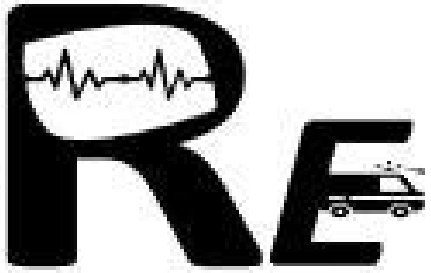
Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Vision

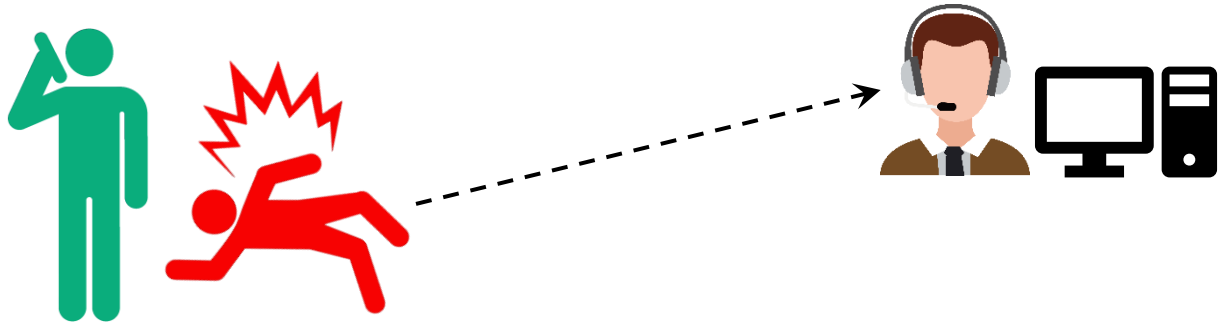


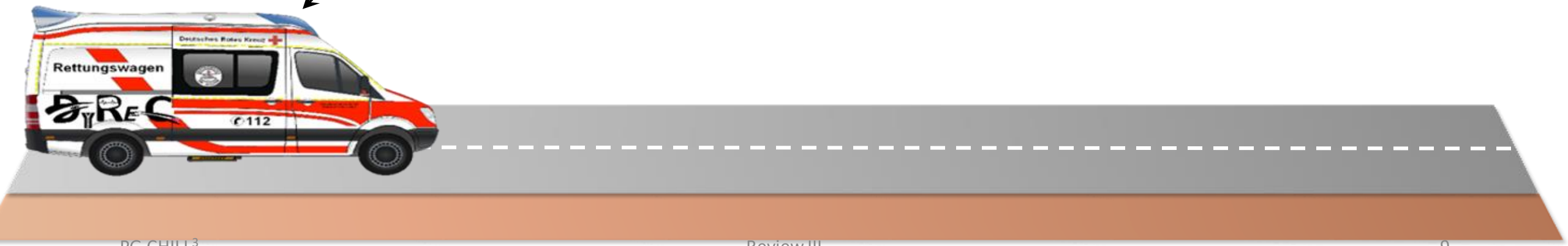
= DYnamic

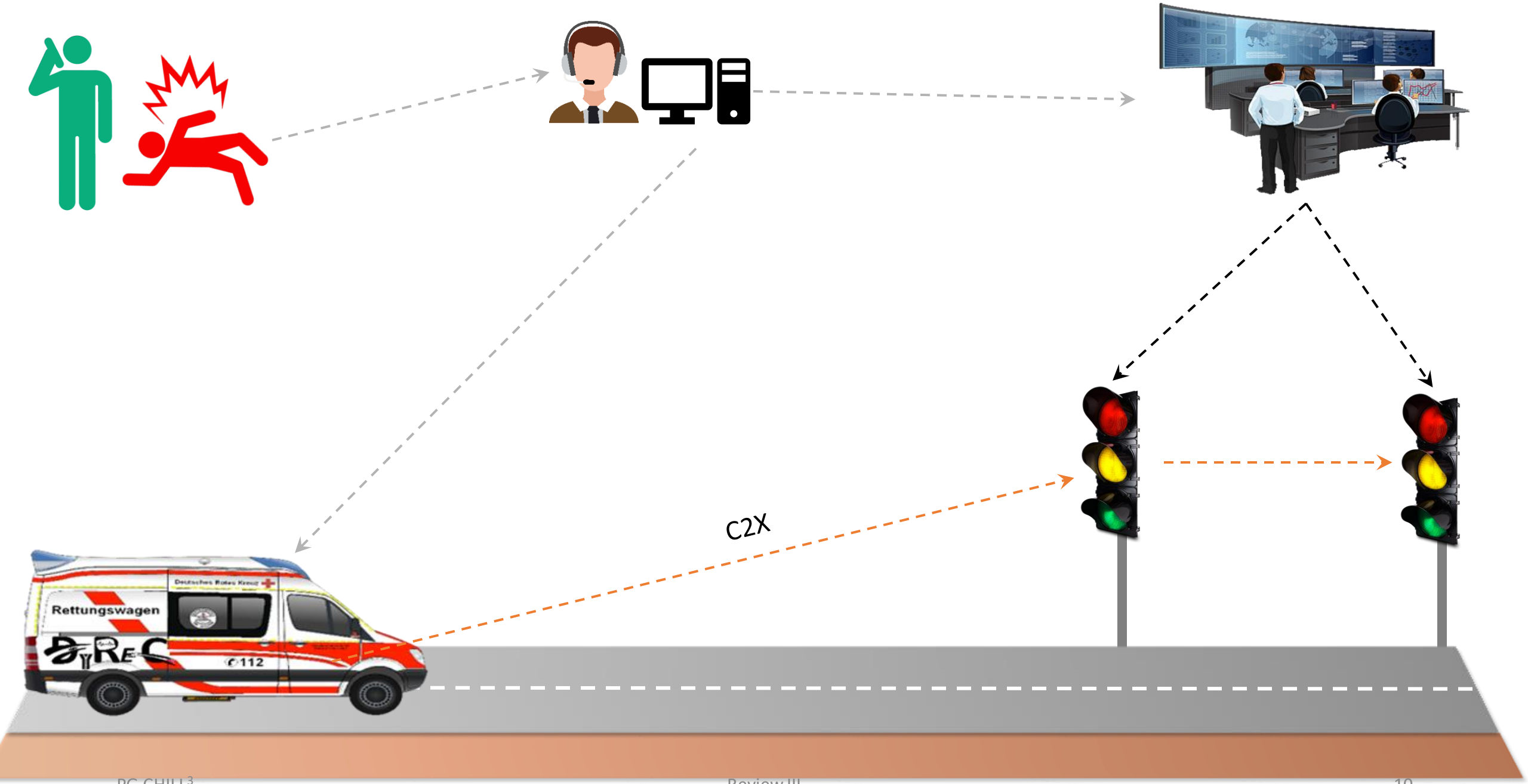


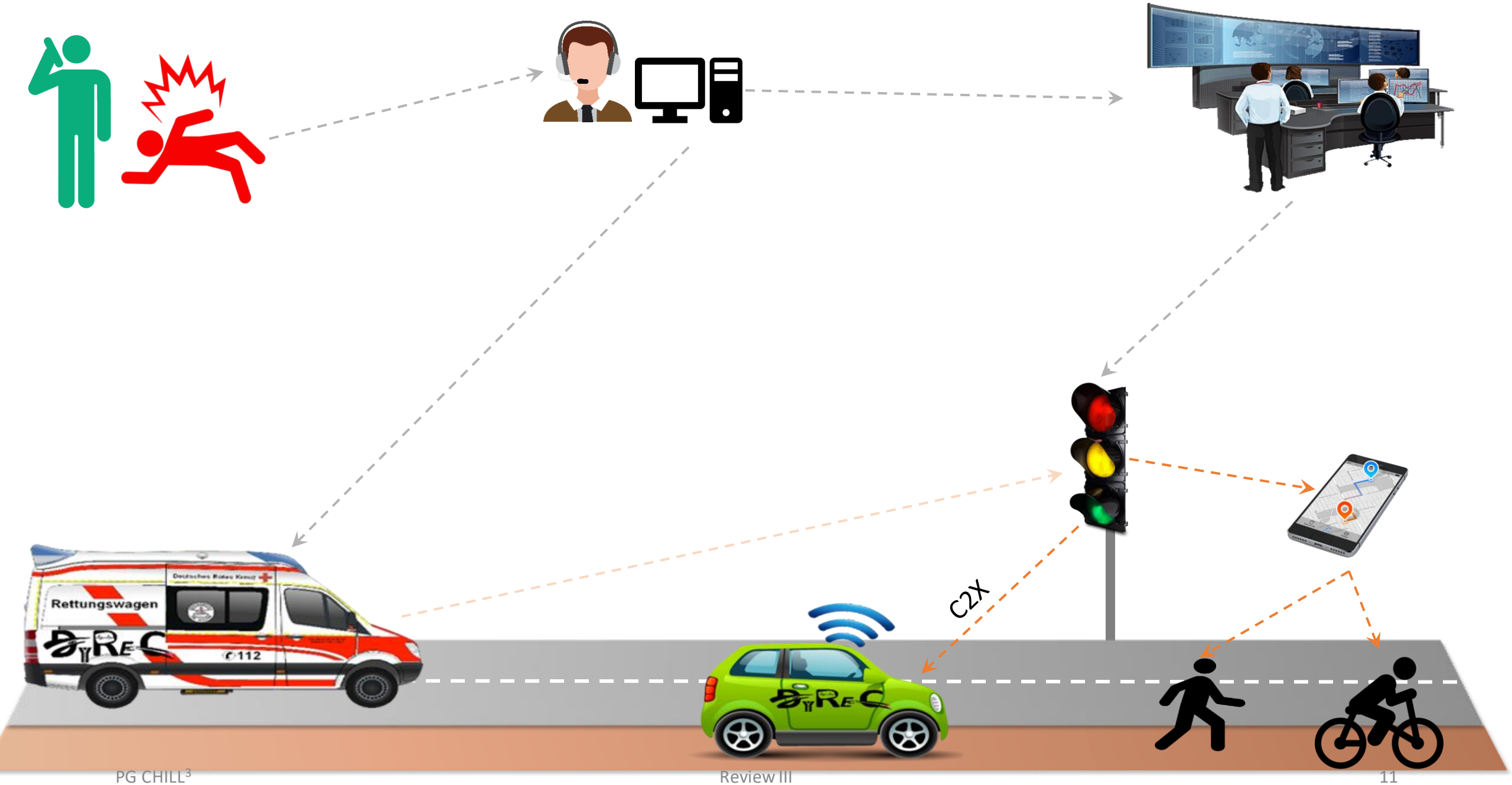
= REscue

= Chill




Rettungswagen
112


C2X
Rettungswagen
112


Rettungswagen
112
C2X

Ziel der Projektgruppe

1. Konzepte für ein DyReC-System entwickeln, die die

- **Hilfsfrist reduzieren und**
- **die Verkehrssicherheit, während einer Alarmfahrt, erhöhen.**

2. Konzepte in einem Demonstrator umsetzen.

- Demonstrator = Simulation von:
  - ELR (inkl. Disponent)
  - Verkehrsmanagementsystemen
  - Kaskadierendes Verfahren (An- und Abmeldung von Rettungsfahrzeugen)
  - Grüne Welle
  - Dynamische Rettungsgasse

Agenda

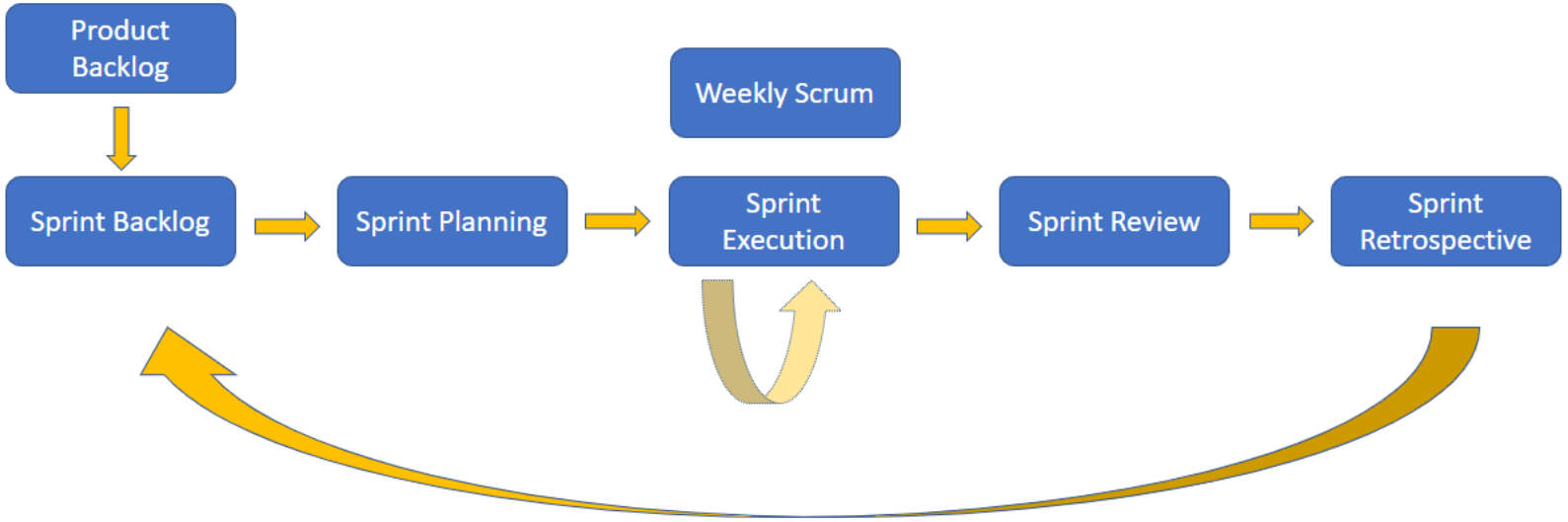
- Vision
- Projektmanagement
  - Vorgehensmodell
  - Zeitplanung
  - Risikomanagement
  - Testmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Agenda

- Vision
- Projektmanagement
  - Vorgehensmodell
  - Zeitplanung
  - Risikomanagement
  - Testmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Vorgehensmodell

- Agil – Scrum
- Iterative Entwicklung in Sprints (Prototypen)
- User Storys
- Jira



Evaluation Vorgehensmodell

Was hat gut funktioniert?

- Aufbau der Prototypen

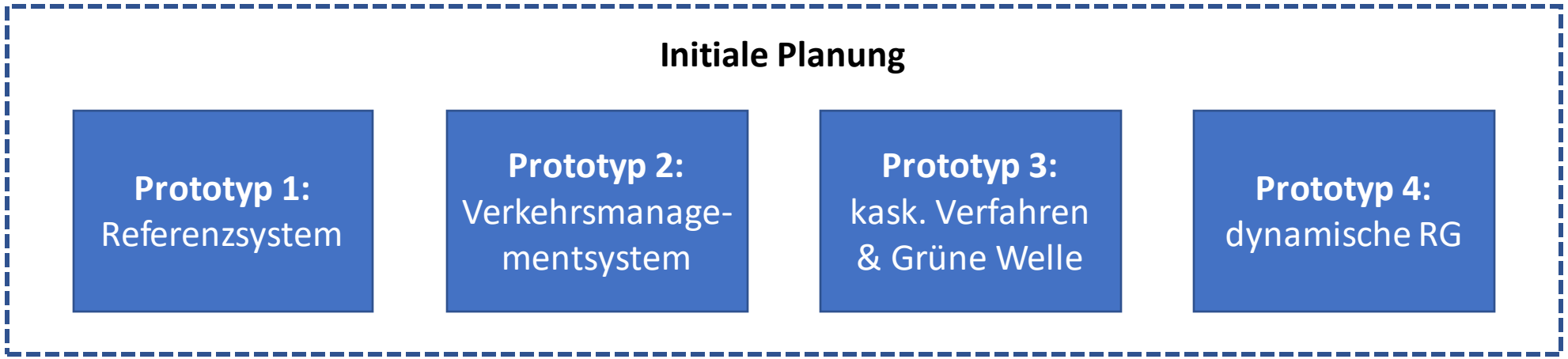
Was haben wir gelernt?

- Kommunikation & Transparenz ist entscheidend
- Einhaltung von Scrum benötigt Commitment & Zeit
- Aufwandschätzungen sind schwierig

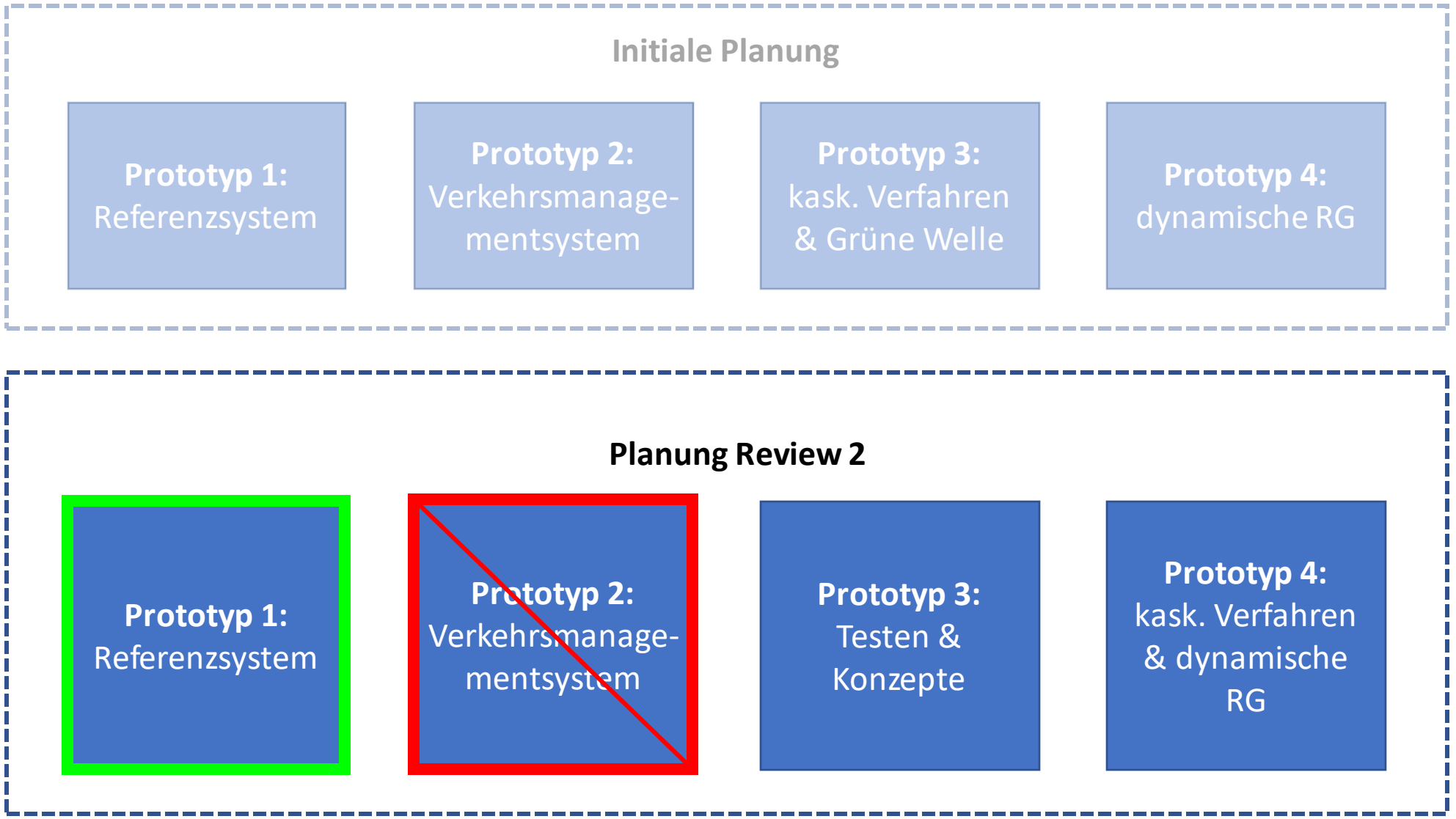
Agenda

- Einleitung & Vision
- Projektmanagement
  - Vorgehensmodell
  - Zeitplanung
  - Risikomanagement
  - Testmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Demonstrator
- Evaluation
- Ausblick
- Fazit

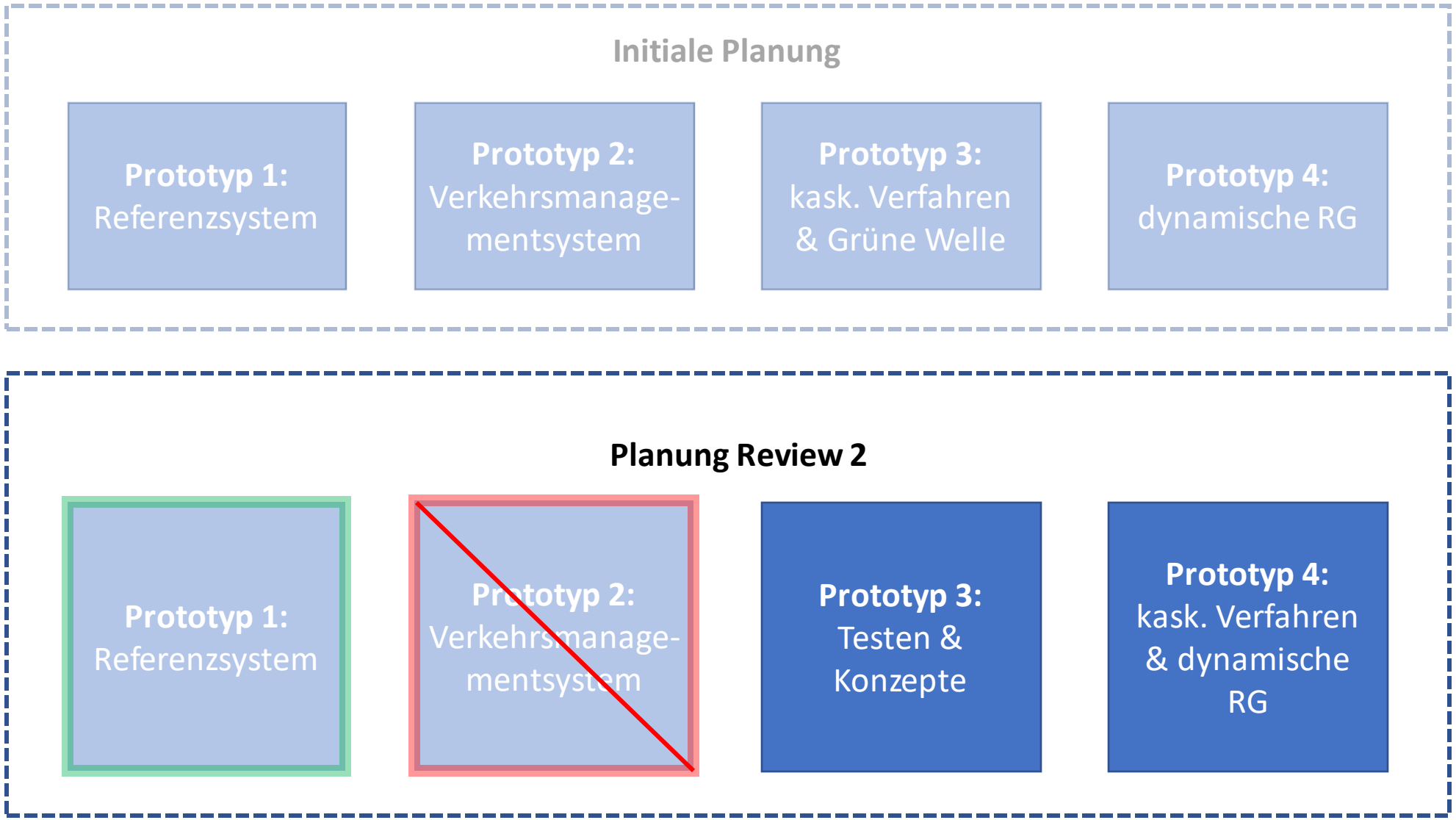
Wiederholung Prototypen



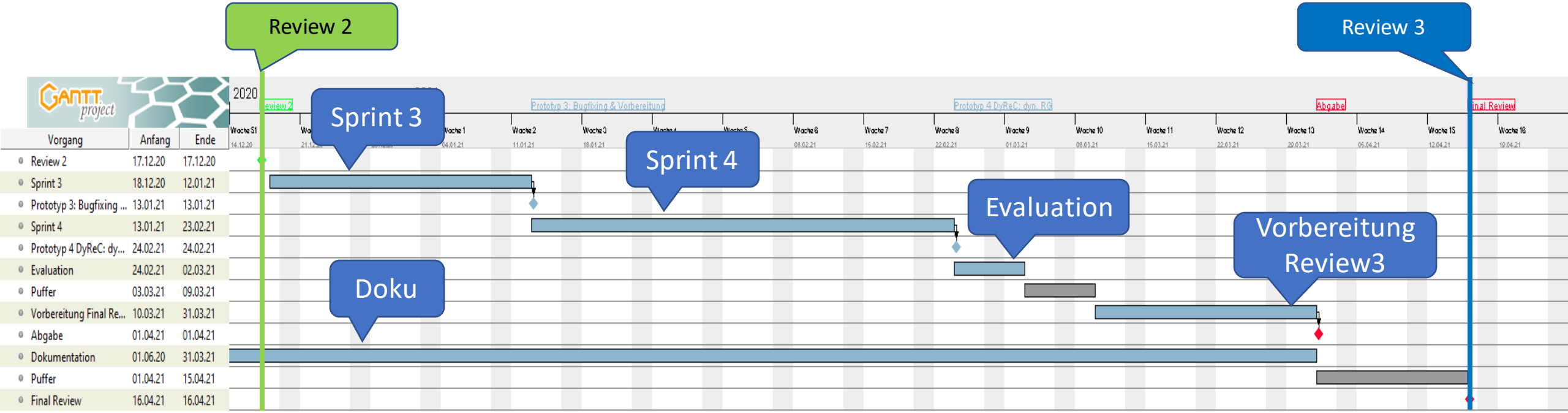
Wiederholung Prototypen



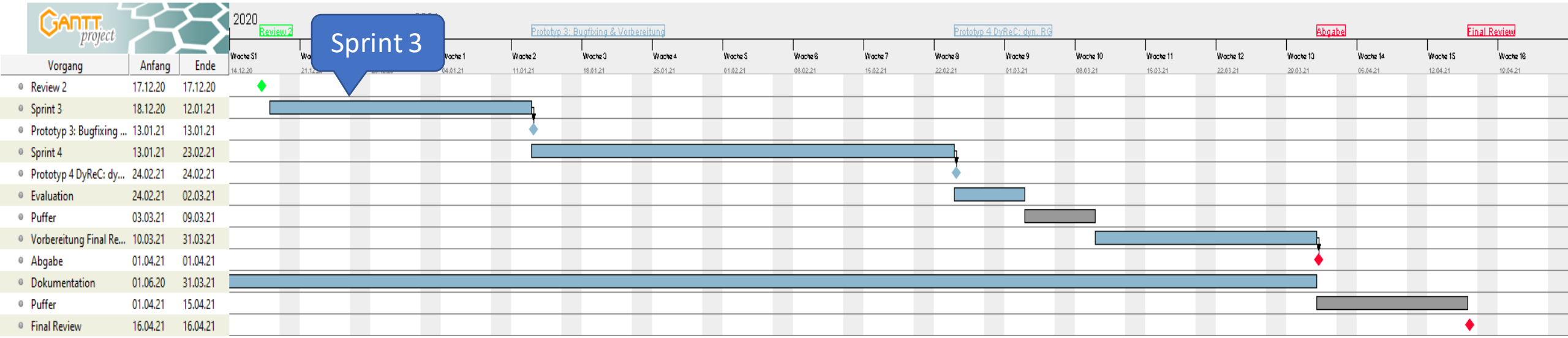
Wiederholung Prototypen



Zeitplanung nach Review 2



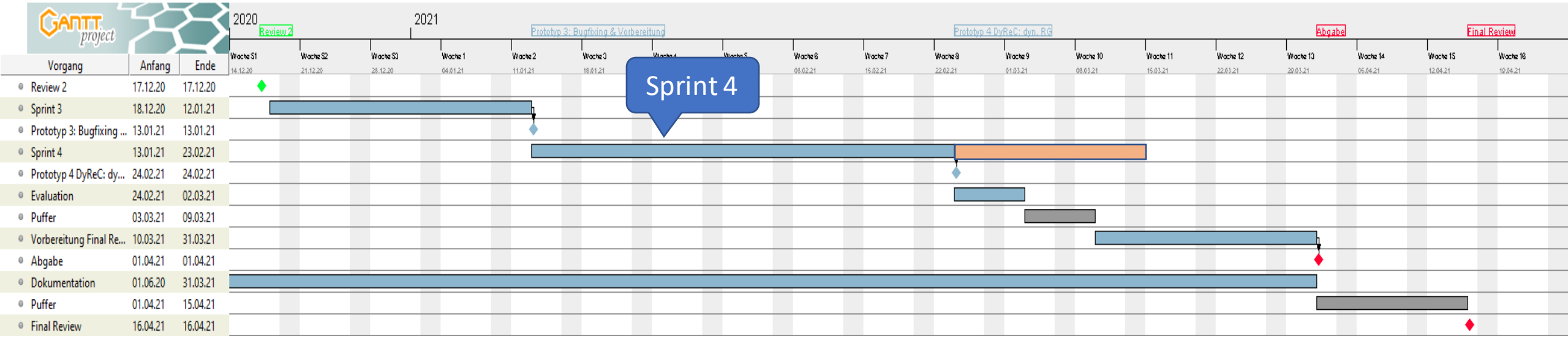
Zeitplanung nach Review 2



Geplant

Verlängert/Verschoben

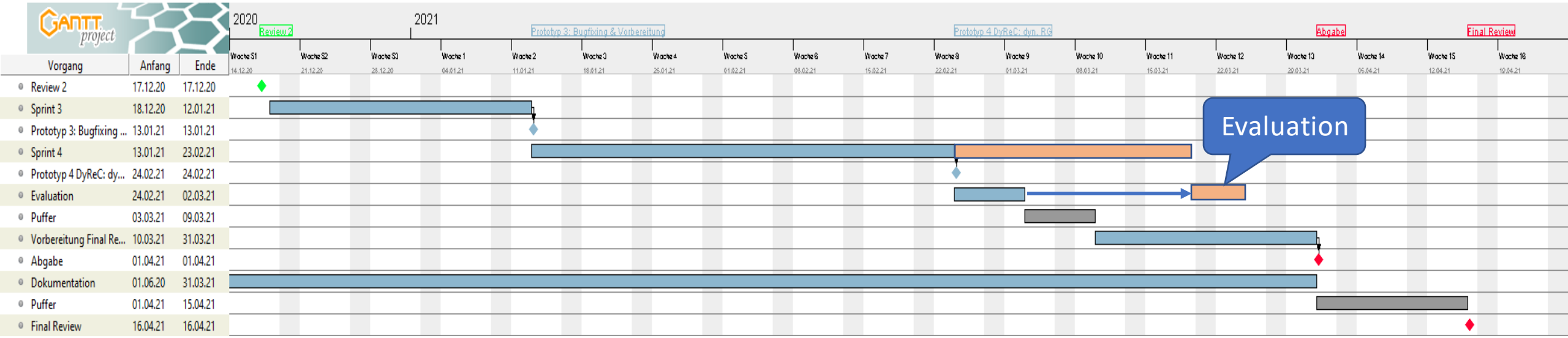
Zeitplanung nach Review 2



Geplant

Verlängert/Verschoben

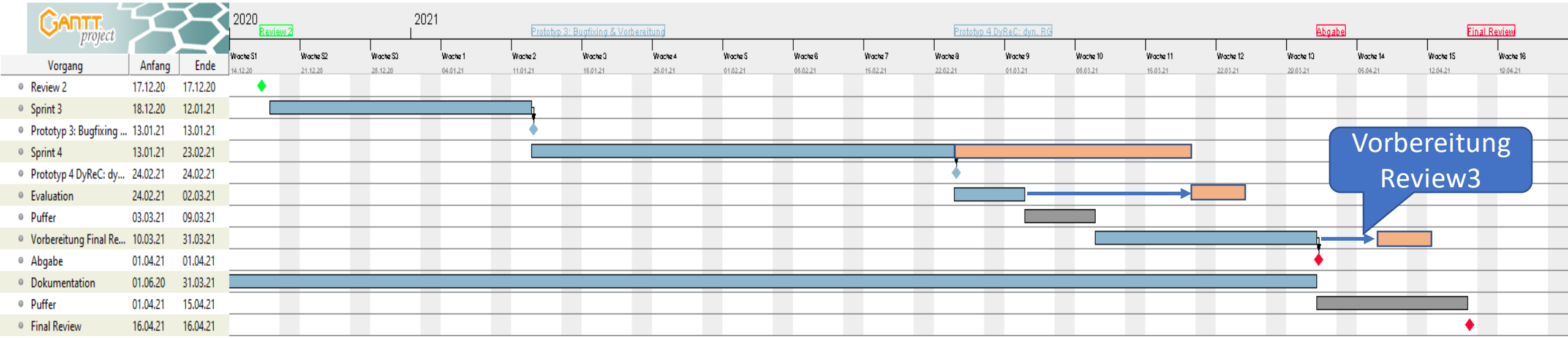
Zeitplanung nach Review 2



Geplant

Verlängert/Verschoben

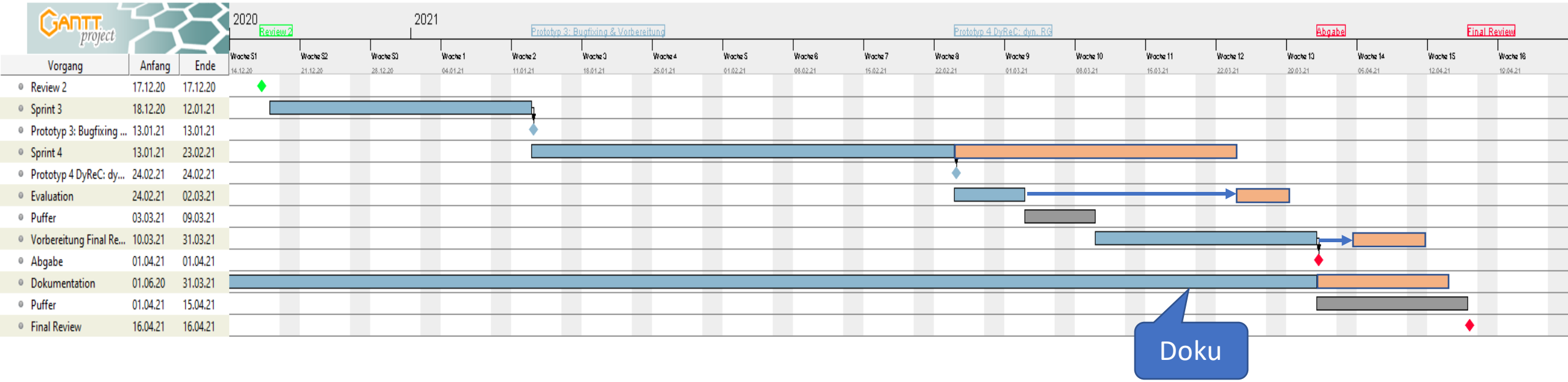
Zeitplanung nach Review 2



Geplant

Verlängert/Verschoben

Zeitplanung nach Review 2



Geplant

Verlängert/Verschoben

Evaluation Zeitplanung

Was hat gut funktioniert?

- Pufferzeiten eingeplant

Was haben wir gelernt?

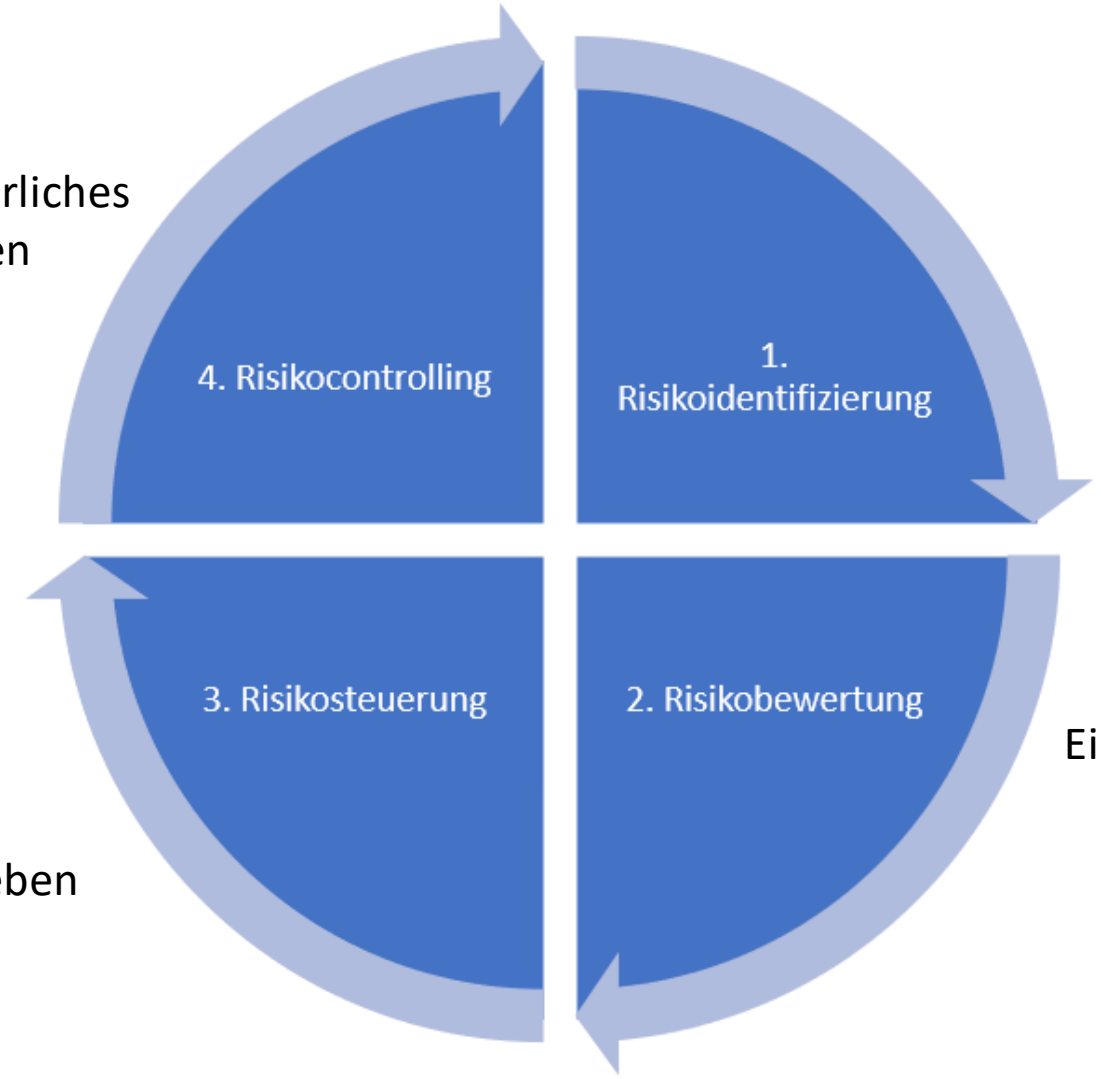
- Ressourcenmanagement ist wichtig für die Zeitplanung
- Einarbeitung benötigt Zeit
- Pufferzeit ist wichtig

Agenda

- Einleitung & Vision
- Projektmanagement
  - Vorgehensmodell
  - Zeitplanung
  - Risikomanagment
  - Testmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Risikomanagement

Kontinuierliches überprüfen

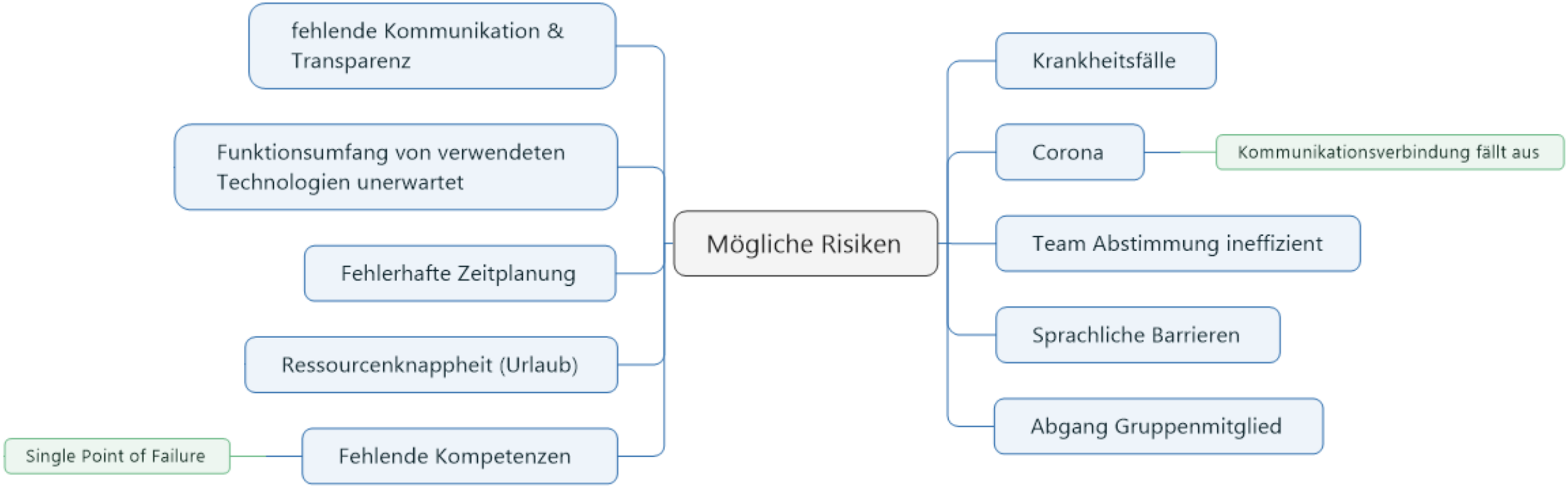


Maßnahmen erheben

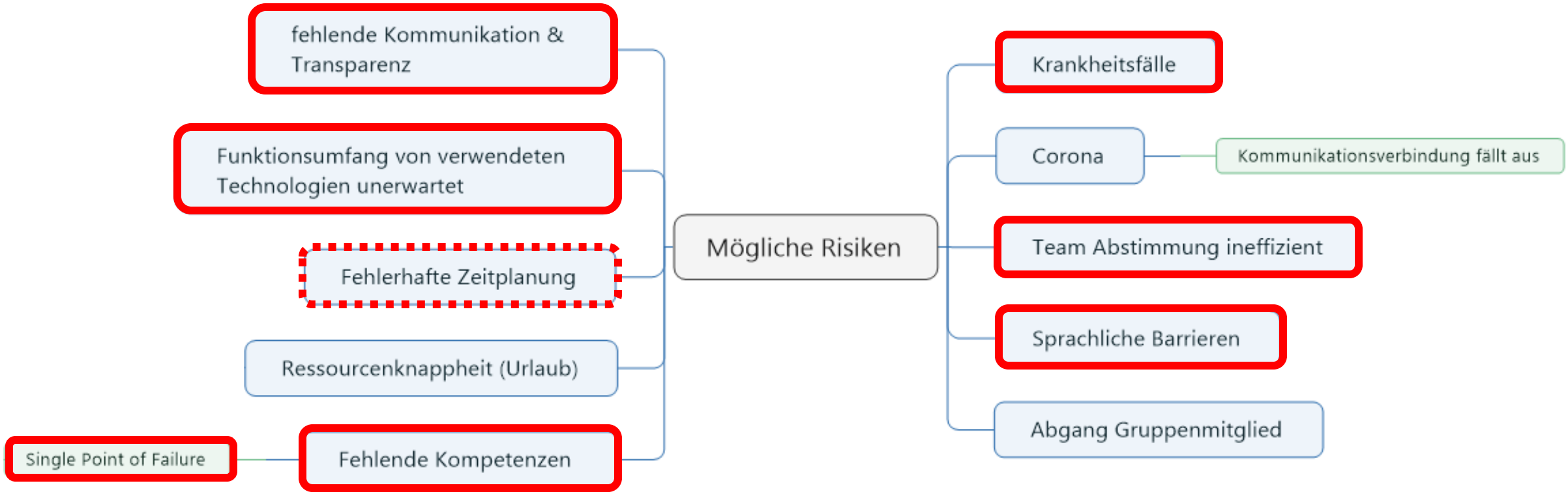
Einteilung in Kategorien:

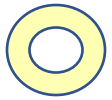
- Niedrig
- Hoch
- Kritisch
- Blocker

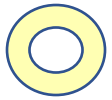
Identifizierte Risiken



Eingetretene Risiken



Risiko	Maßnahmen	Nutzen
Team Abstimmungen ineffizient	Anschaffung Zeichentabletts	
Fehlende Kompetenzen	Pair Programming	
	Fragen an Team-Mitglieder	
Fehlende Kommunikation & Transparenz	Jira	
Single Point of Failure	-	

Risiko	Maßnahmen	Nutzen
Abstimmungen ineffizient	Anschaffung Zeichentabletts	
	Entwicklungsleitung - Trennung Arbeits- und Teammeeting	
Fehlende Kompetenzen	Pair Programming	
	Fragen an Team-Mitglieder	
Fehlende Kommunikation & Transparenz	Jira	
Single Point of Failure	-	

Risiko	Maßnahmen	Nutzen
Abstimmungen ineffizient	Anschaffung Zeichentabletts	
	Entwicklungsleitung - Trennung Arbeits- und Teammeeting	
Fehlende Kompetenzen	Pair Programming	
	Fragen an Team-Mitglieder	
Fehlende Kommunikation & Transparenz	Jira	
Single Point of Failure	-	

Risiko	Maßnahmen	Nutzen
Abstimmungen ineffizient	Anschaffung Zeichentabletts	
	Entwicklungsleitung - Trennung Arbeits- und Teammeeting	
Fehlende Kompetenzen	**Entwicklungsleitung** - Pair Programming	
	Entwicklungsleitung - Fragen an Team-Mitglieder	
Fehlende Kommunikation & Transparenz	Jira	
Single Point of Failure	-	

Risiko	Maßnahmen	Nutzen
Abstimmungen ineffizient	Anschaffung Zeichentabletts	
	Entwicklungsleitung - Trennung Arbeits- und Teammeeting	
Fehlende Kompetenzen	Entwicklungsleitung - Pair Programming	
	Entwicklungsleitung - Fragen an Team-Mitglieder	
Fehlende Kommunikation & Transparenz	Jira	
Single Point of Failure	-	

Risiko	Maßnahmen	Nutzen
Abstimmungen ineffizient	Anschaffung Zeichentabletts	
	Entwicklungsleitung - Trennung Arbeits- und Teammeeting	
Fehlende Kompetenzen	Entwicklungsleitung - Pair Programming	
	Entwicklungsleitung - Fragen an Team-Mitglieder	
Fehlende Kommunikation & Transparenz	**Feingranulare Arbeitspakete in** Jira	
Single Point of Failure	-	

Risiko	Maßnahmen	Nutzen
Abstimmungen ineffizient	Anschaffung Zeichentabletts	
	Entwicklungsleitung - Trennung Arbeits- und Teammeeting	
Fehlende Kompetenzen	Entwicklungsleitung - Pair Programming	
	Entwicklungsleitung - Fragen an Team-Mitglieder	
Fehlende Kommunikation & Transparenz	Feingranulare Arbeitspakete in Jira	
Single Point of Failure	-	

Risiko	Maßnahmen	Nutzen
Abstimmungen ineffizient	Anschaffung Zeichentabletts	
	Entwicklungsleitung - Trennung Arbeits- und Teammeeting	
Fehlende Kompetenzen	Entwicklungsleitung - Pair Programming	
	Entwicklungsleitung - Fragen an Team-Mitglieder	
Fehlende Kommunikation & Transparenz	Feingranulare Arbeitspakete in Jira	
Single Point of Failure	**Entwicklungsleitung**	

Evaluation Risikomanagement

Was hat gut funktioniert?

- Risiken konnten identifiziert werden
- Aufdecken der Risiken durch das Team
- Maßnahmen konnten teilweise umgesetzt werden

Was haben wir gelernt?

- Kommunikation & Transparenz ist entscheidend
- Frühzeitiges Erkennen von Risiken ist viel Wert
- Konkrete Gegenmaßnahmen zu Entwickeln ist herausfordernd
- Kontinuierliches Risikomanagement ist zeitintensiv

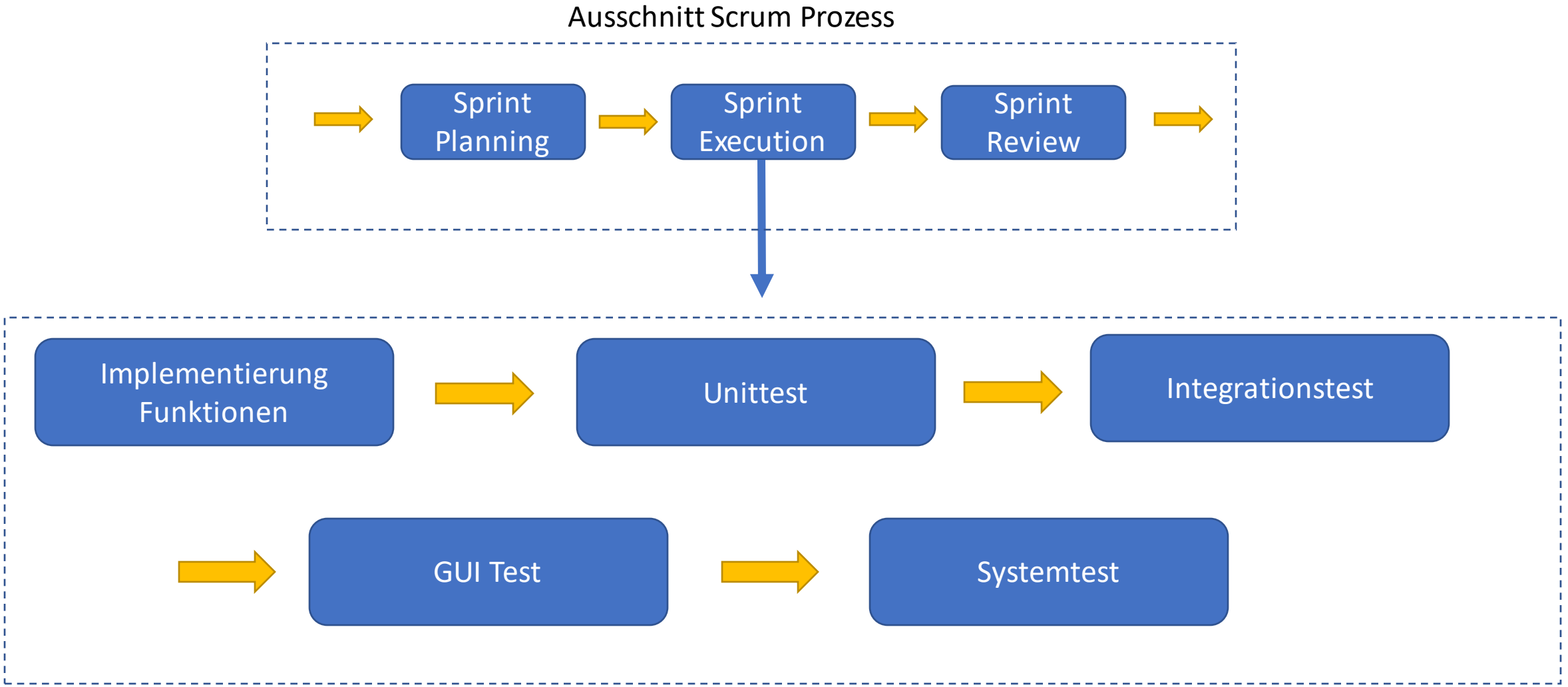
Agenda

- Einleitung & Vision
- Projektmanagement
  - Vorgehensmodell
  - Zeitplanung
  - Risikomanagement
  - Testmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Testarten

- Unittest
  - Isoliertes, automatisiertes Testen von einzelnen Methoden
  - Abwägen ob Aufwand zum Mocking im Verhältnis zum Mehrwert steht
- Integrationstest
  - Automatisiertes Testen des Zusammenspiels von Methoden
  - Methoden sollten zuvor durch Unittest abgedeckt sein
- GUI-Test
  - Manuelles Testen der GUI
- Systemtest
  - Testen des Gesamtsystems

Testdurchführung - Planung



Testdurchführung - Realität

- Sprint 3 wurde zu einem Testsprint (& Konzepte)
  - Entwicklung & Dokumentation von Unit- und Integrationstest
  - GUI Test
  - Bugfixing

- Zum Ende von Sprint 4
  - Unittest und Integrationstest erweitert
  - GUI Test
  - Systemtest

Evaluation Testmanagement

Was hat gut funktioniert?

- Technisches Verständnis in Bezug auf DyReC erweitern, durch Trennung Entwickler & Tester

Was haben wir gelernt?

- Testen muss fest eingeplant werden
- Aussagekraft von Tests hängt von Testumgebung ab

Fazit Projektmanagement

- Was haben wir gelernt?
  - Vorgehensmodelle sind komplex und müssen individuell auf die Rahmenbedingung angepasst werden.
  - Eine übergeordnete Rolle ist sinnvoll, um den Gesamtüberblick zu behalten.
  - Projektmanagement ist der Schlüssel für ein erfolgreiches Projekt.
  - Projektmanagement ist aufwändig.

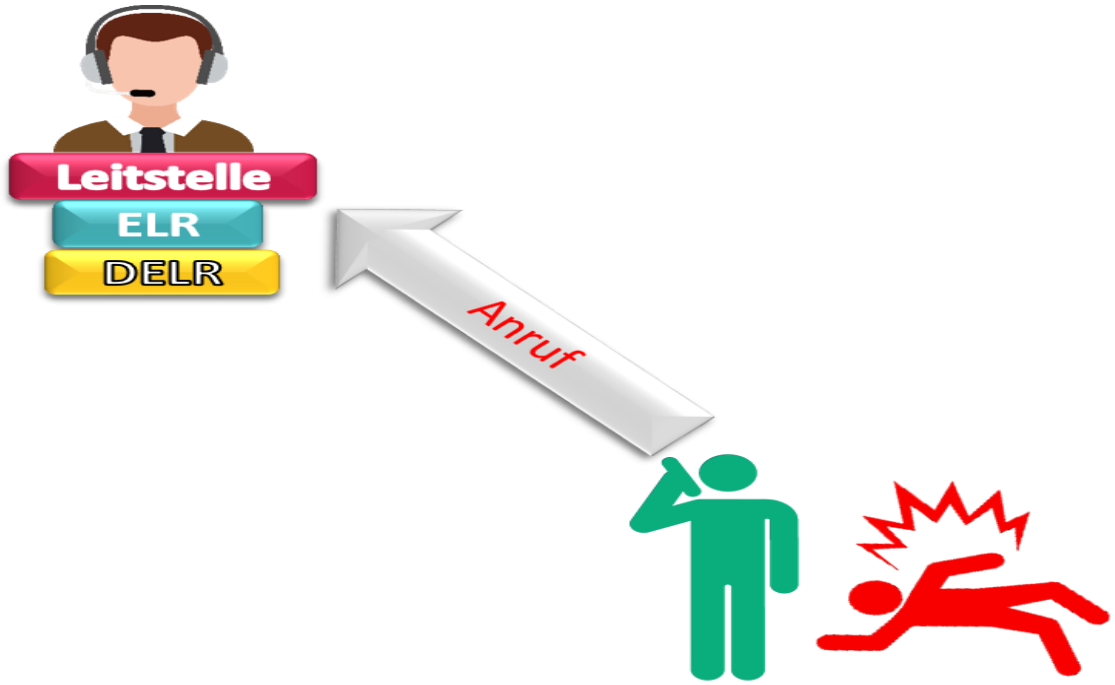
Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Test
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Agenda

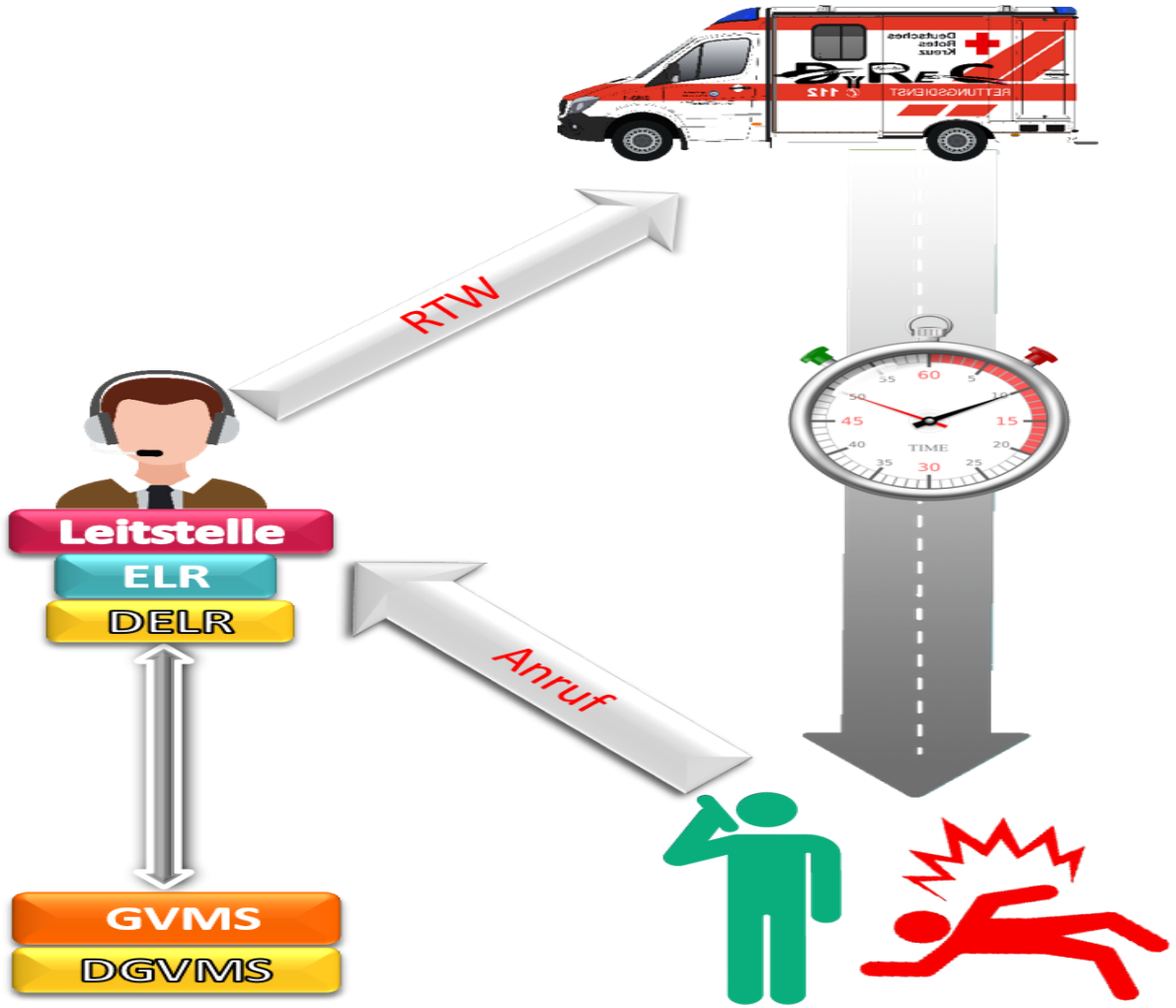
- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Test
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Initialer Systemaufbau



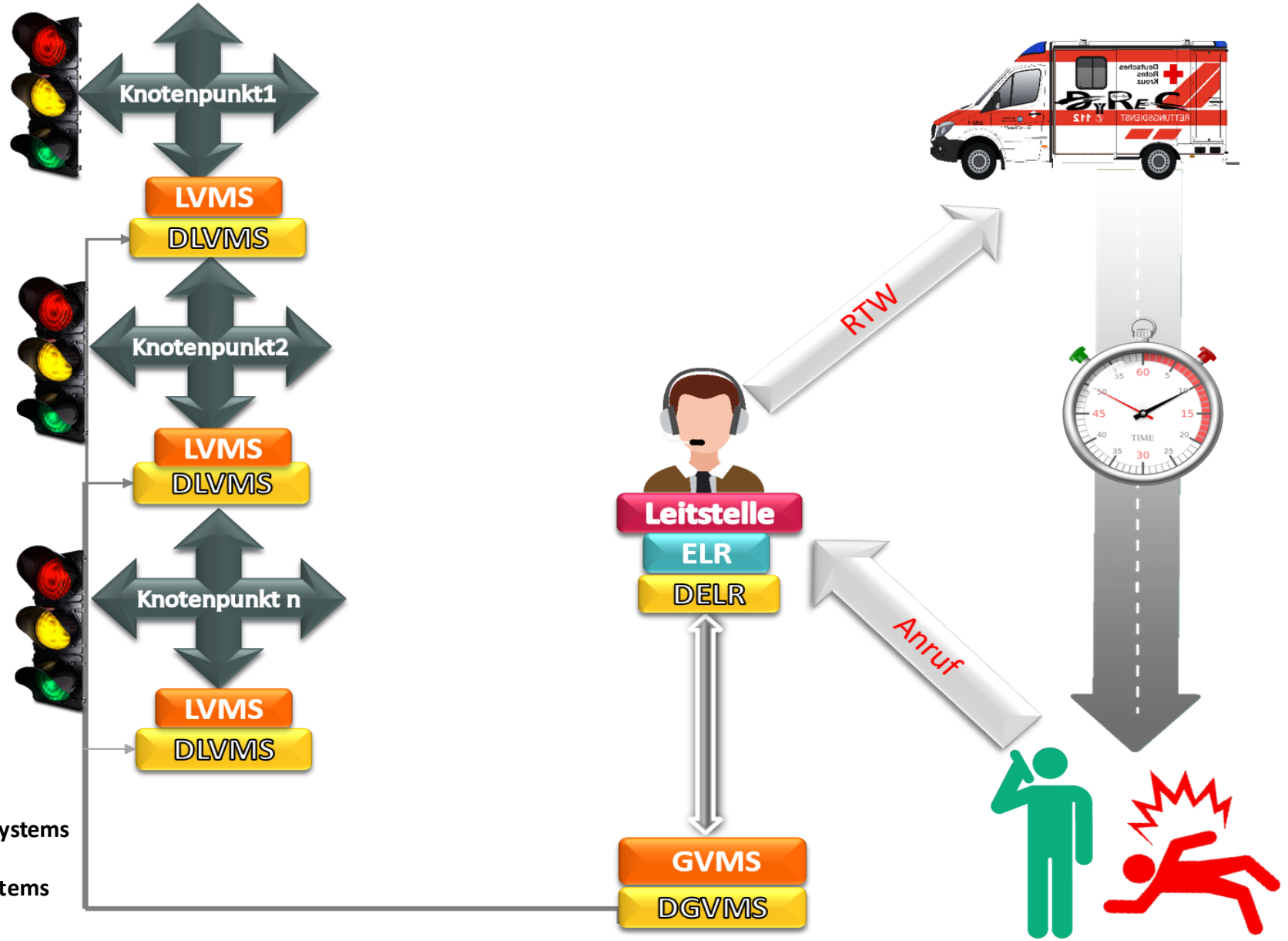
DELR= DyReC-System des Einsatzleitrechners

Initialer Systemaufbau



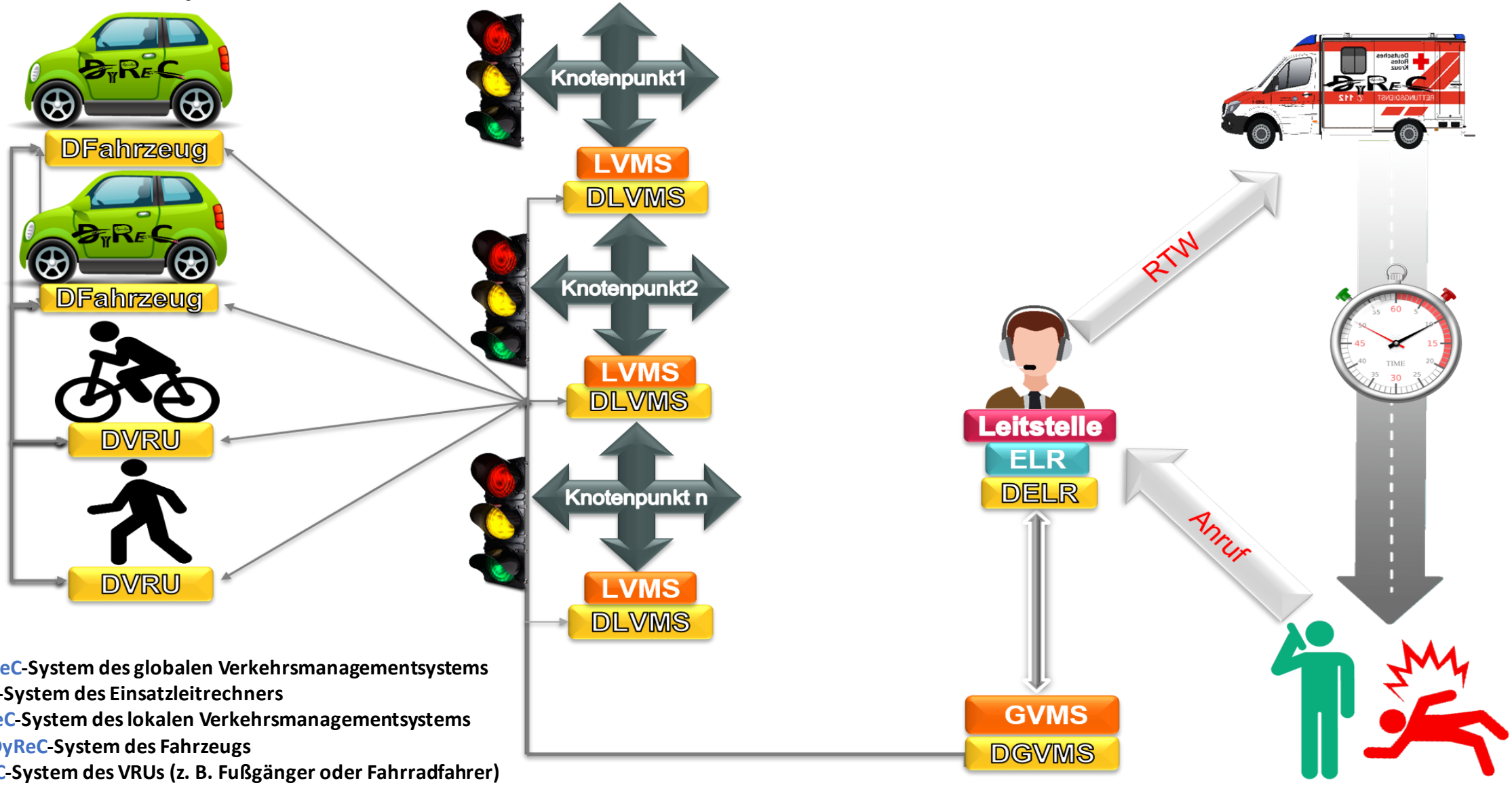
DELR= DyReC-System des Einsatzleitrechners
DGVMS= DyReC-System des globalen Verkehrsmanagementsystems

Initialer Systemaufbau



DGVMS= DyReC-System des globalen Verkehrsmanagementsystems
DELR= DyReC-System des Einsatzleitrechners
DLVMS= DyReC-System des lokalen Verkehrsmanagementsystems

Initialer Systemaufbau



DGVMS= DyReC-System des globalen Verkehrsmanagementsystems
DELR= DyReC-System des Einsatzleitrechners
DLVMS= DyReC-System des lokalen Verkehrsmanagementsystems
DFahrzeug= DyReC-System des Fahrzeugs
DVRU= DyReC-System des VRUs (z. B. Fußgänger oder Fahrradfahrer)

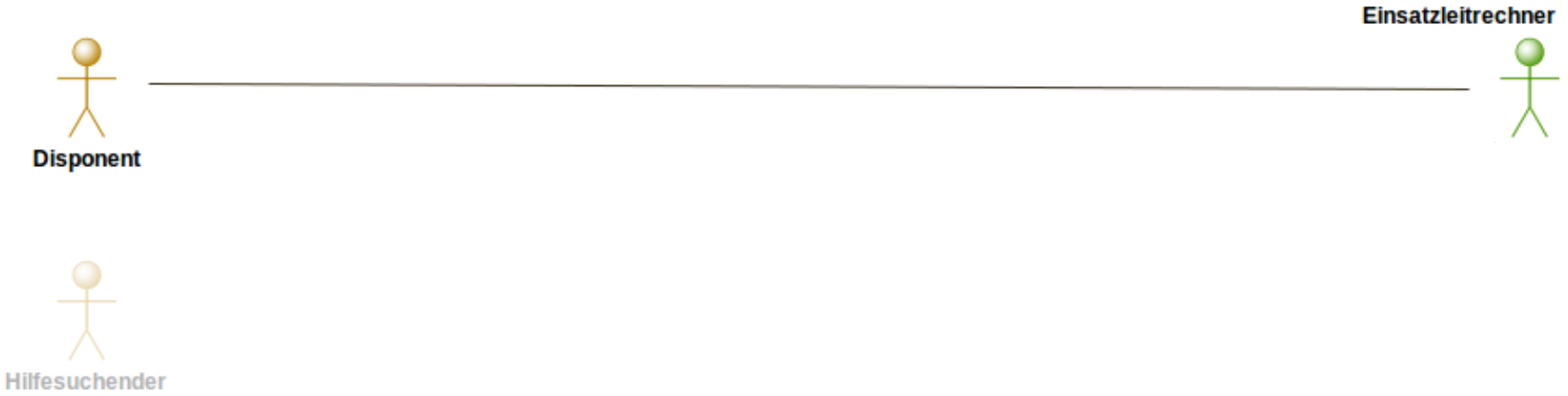
Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Test
- Demonstrator
- Evaluation
- Ausblick
- Fazit

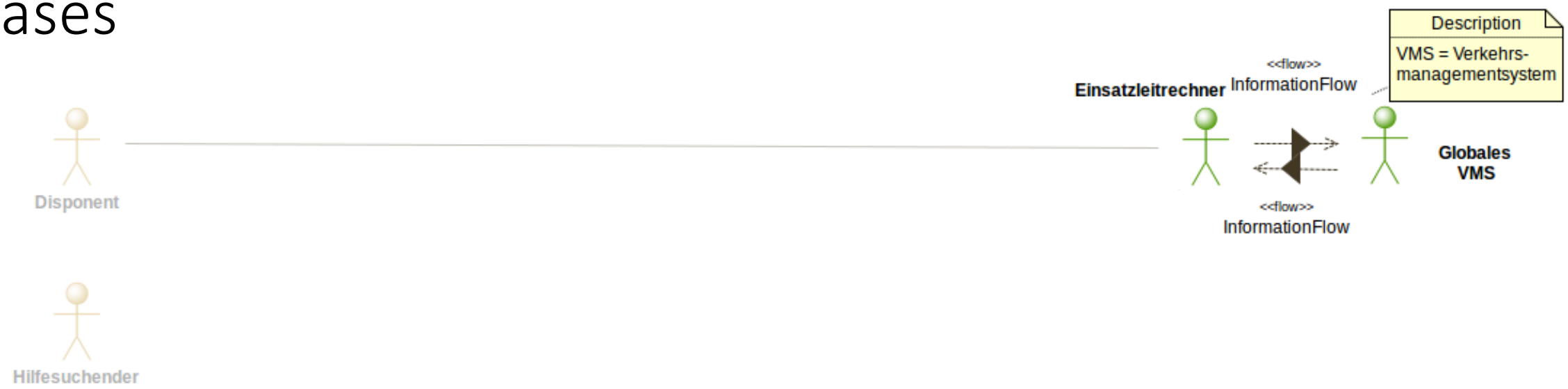
Use Cases

Use Cases

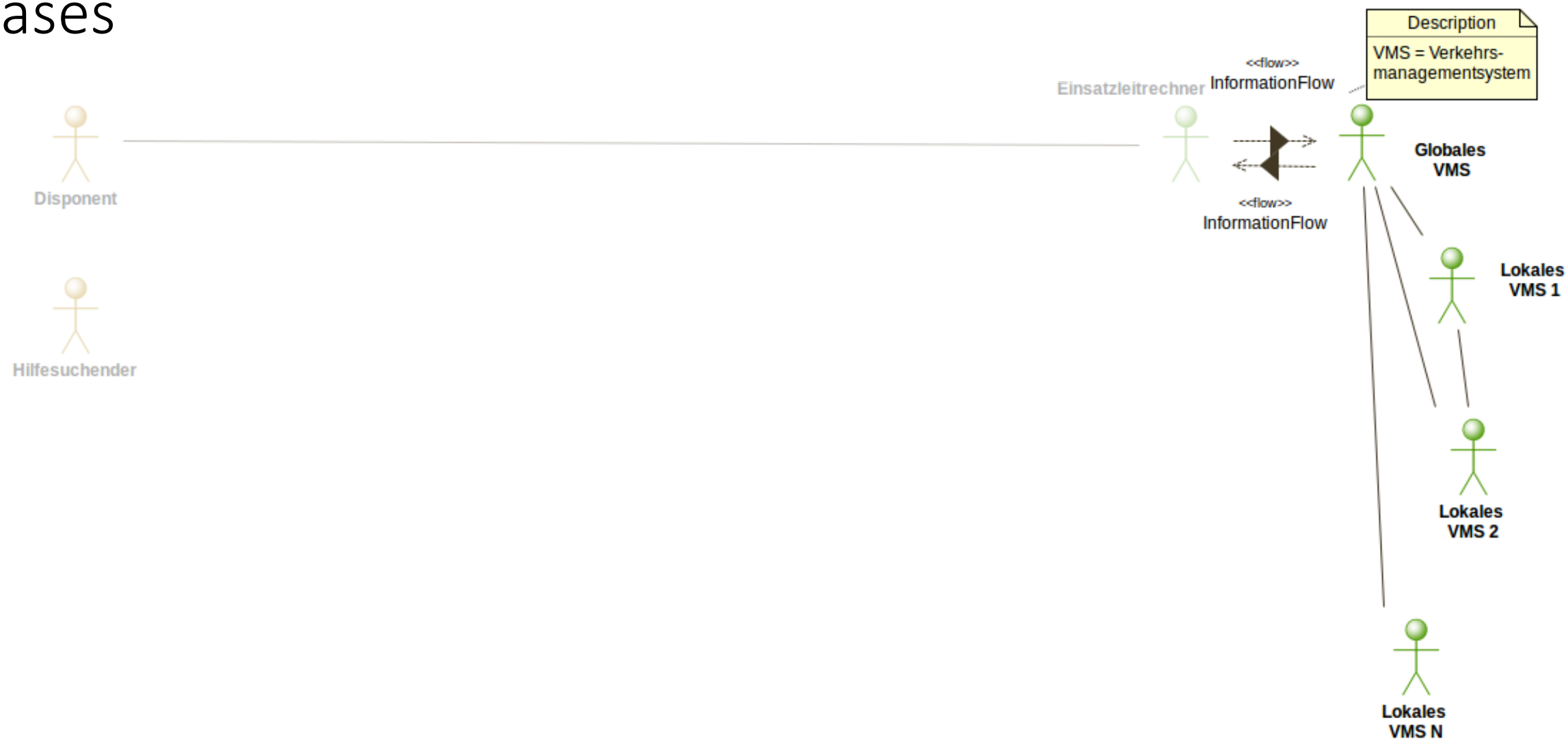
Use Cases



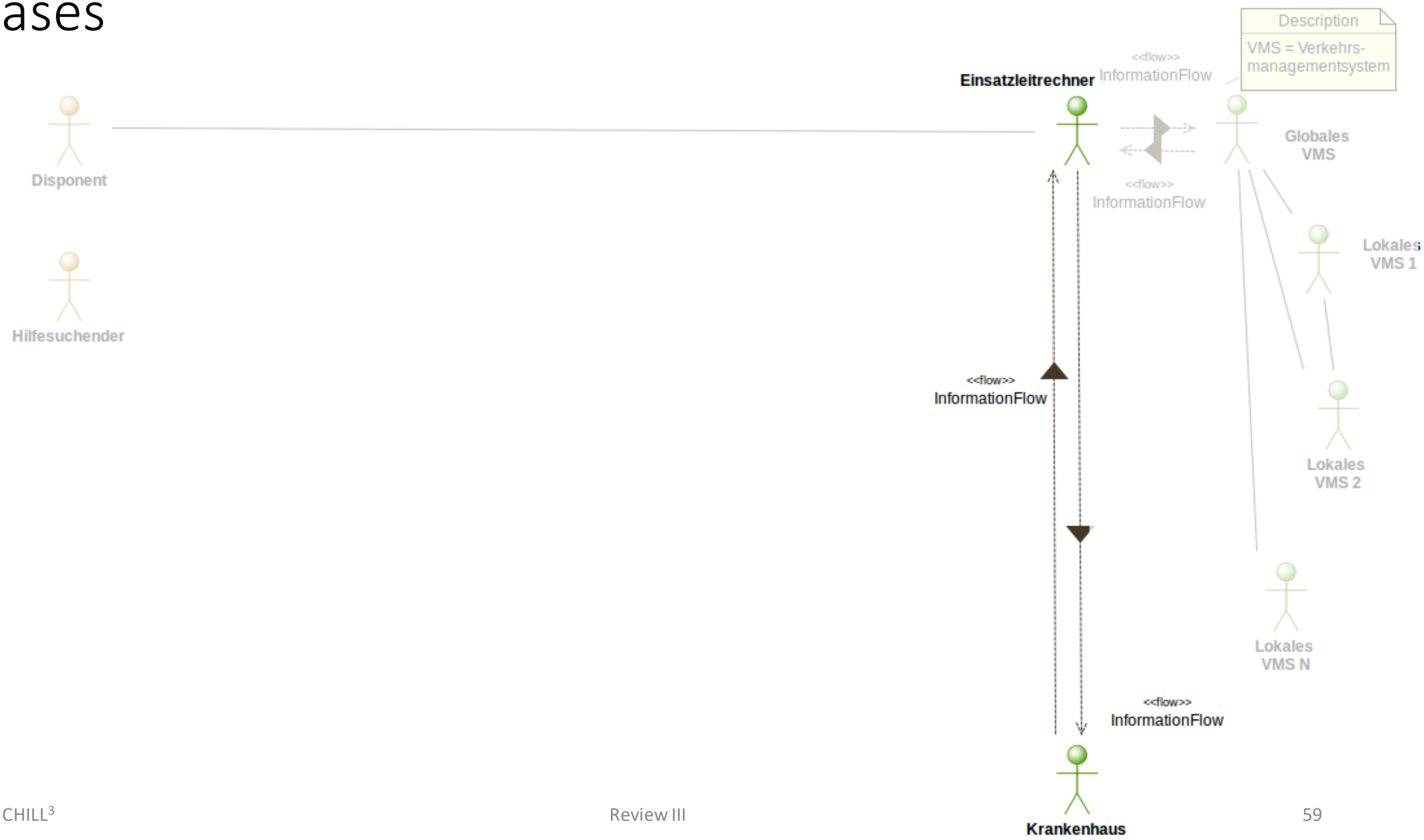
Use Cases


Description
VMS = Verkehrs-managementsystem
<<flow>>
InformationFlow
Einsatzleitrechner
Globales VMS
<<flow>>
InformationFlow
Disponent
Hilfesuchender

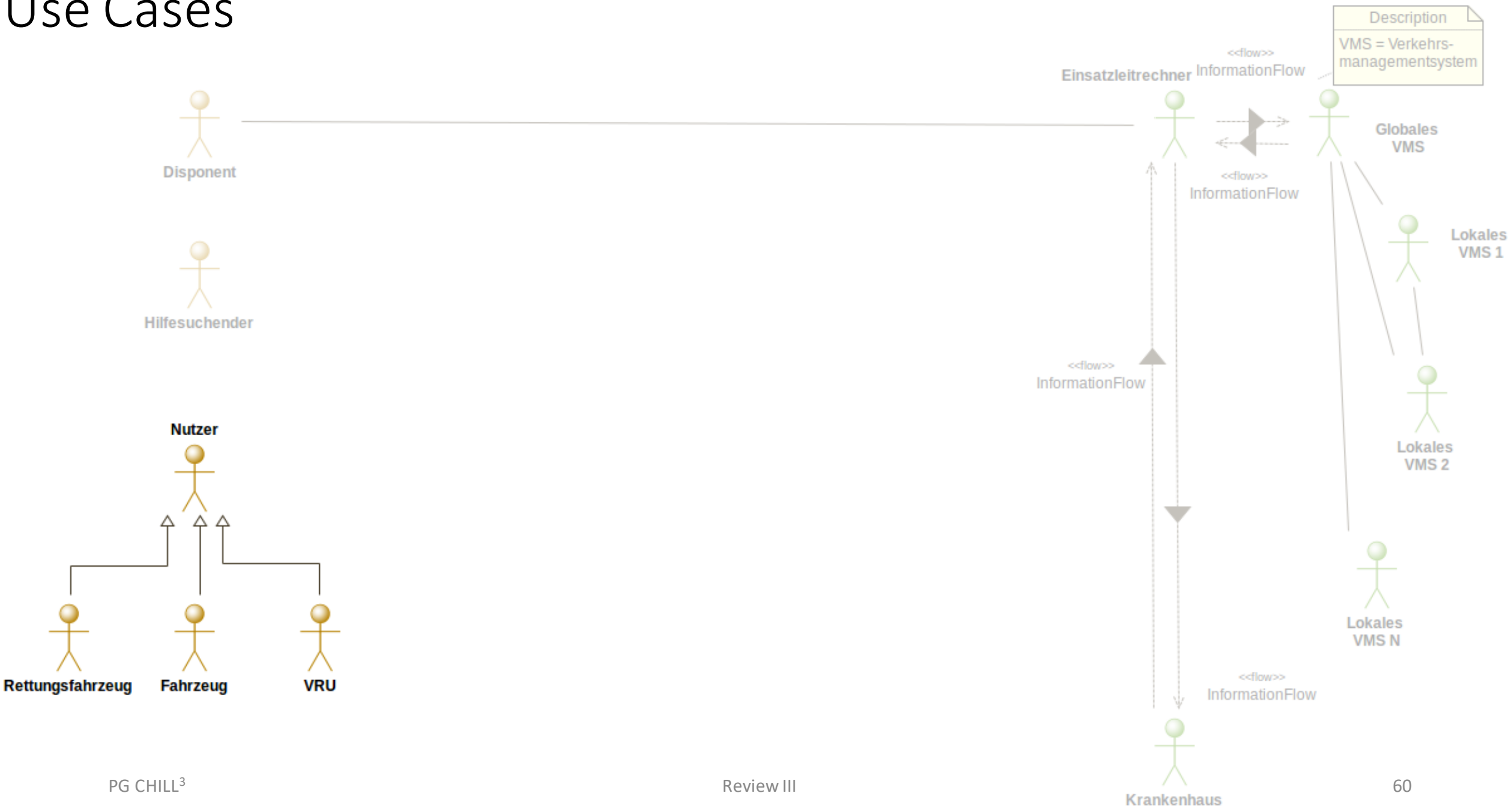
Use Cases


Description
VMS = Verkehrs-managementsystem
Einsatzleitrechner
<<flow>>
InformationFlow
<<flow>>
InformationFlow
Disponent
Hilfesuchender
Globales VMS
Lokales VMS 1
Lokales VMS 2
Lokales VMS N

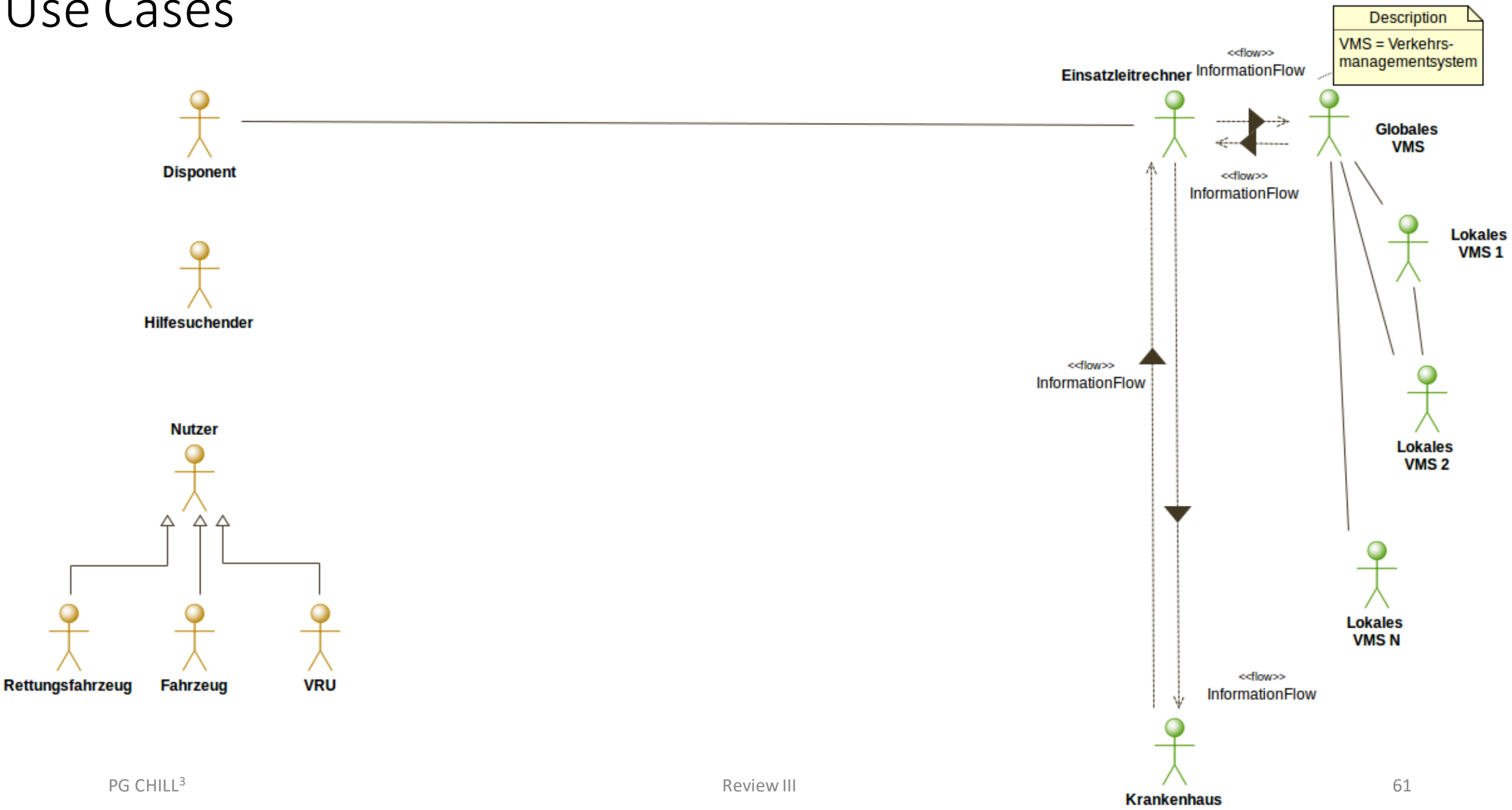
Use Cases


Description
VMS = Verkehrs-managementsystem
Disponent
Hilfesuchender
Einsatzleitrechner
<<flow>>
InformationFlow
<<flow>>
InformationFlow
Globales VMS
Lokales VMS 1
Lokales VMS 2
Lokales VMS N
<<flow>>
InformationFlow
<<flow>>
InformationFlow
Krankenhaus

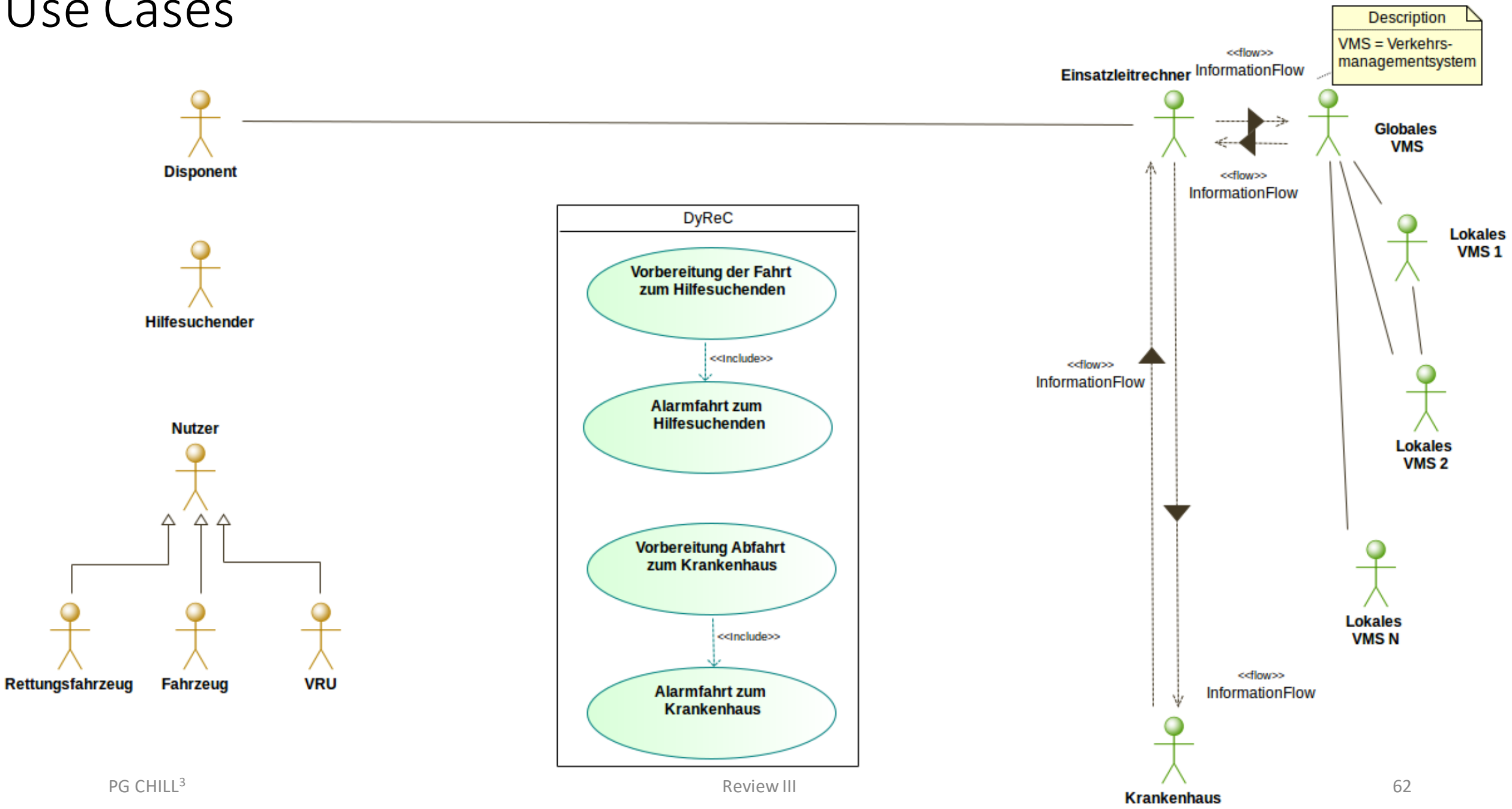
Use Cases


Description
VMS = Verkehrs-
managementsystem
Disponent
Hilfesuchender
Einsatzleitrechner
<<flow>>
InformationFlow
Globales
VMS
<<flow>>
InformationFlow
Lokales
VMS 1
Lokales
VMS 2
<<flow>>
InformationFlow
Lokales
VMS N
<<flow>>
InformationFlow
Krankenhaus
Nutzer
Rettungsfahrzeug
Fahrzeug
VRU

Use Cases

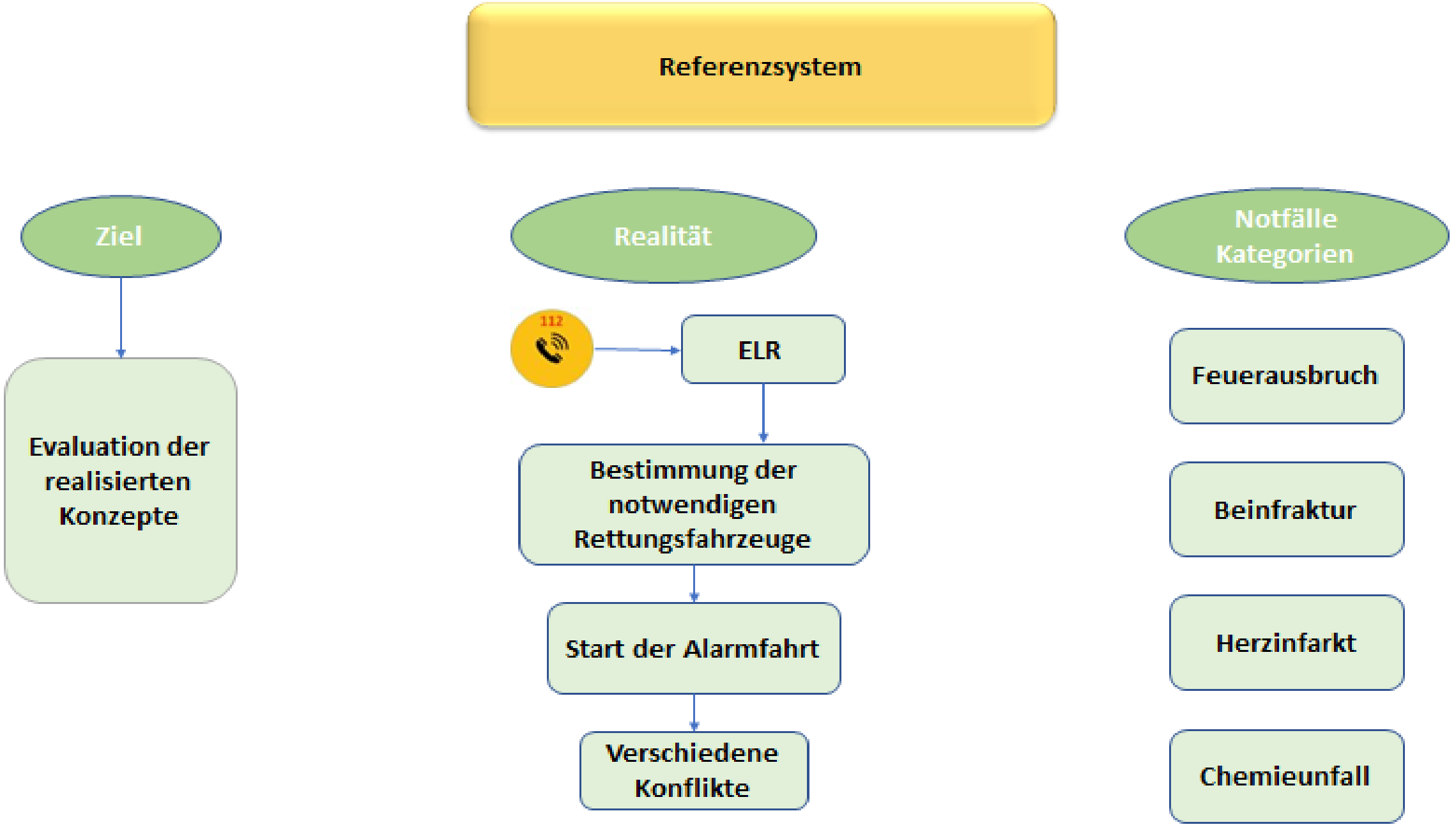

Description
VMS = Verkehrs-managementsystem
Disponent
Hilfesuchender
Nutzer
Rettungsfahrzeug
Fahrzeug
VRU
Einsatzleitrechner
<<flow>>
InformationFlow
<<flow>>
InformationFlow
<<flow>>
InformationFlow
<<flow>>
InformationFlow
Globales VMS
Lokales VMS 1
Lokales VMS 2
Lokales VMS N
Krankenhaus

Use Cases


Disponent
Hilfesuchender
Nutzer
Rettungsfahrzeug
Fahrzeug
VRU
DyReC
Vorbereitung der Fahrt zum Hilfesuchenden
<<Include>>
Alarmfahrt zum Hilfesuchenden
Vorbereitung Abfahrt zum Krankenhaus
<<Include>>
Alarmfahrt zum Krankenhaus
Description
VMS = Verkehrs-managementsystem
Einsatzleitrechner
<<flow>>
InformationFlow
Globales VMS
<<flow>>
InformationFlow
Lokales VMS 1
Lokales VMS 2
Lokales VMS N
<<flow>>
InformationFlow
<<flow>>
InformationFlow
Krankenhaus

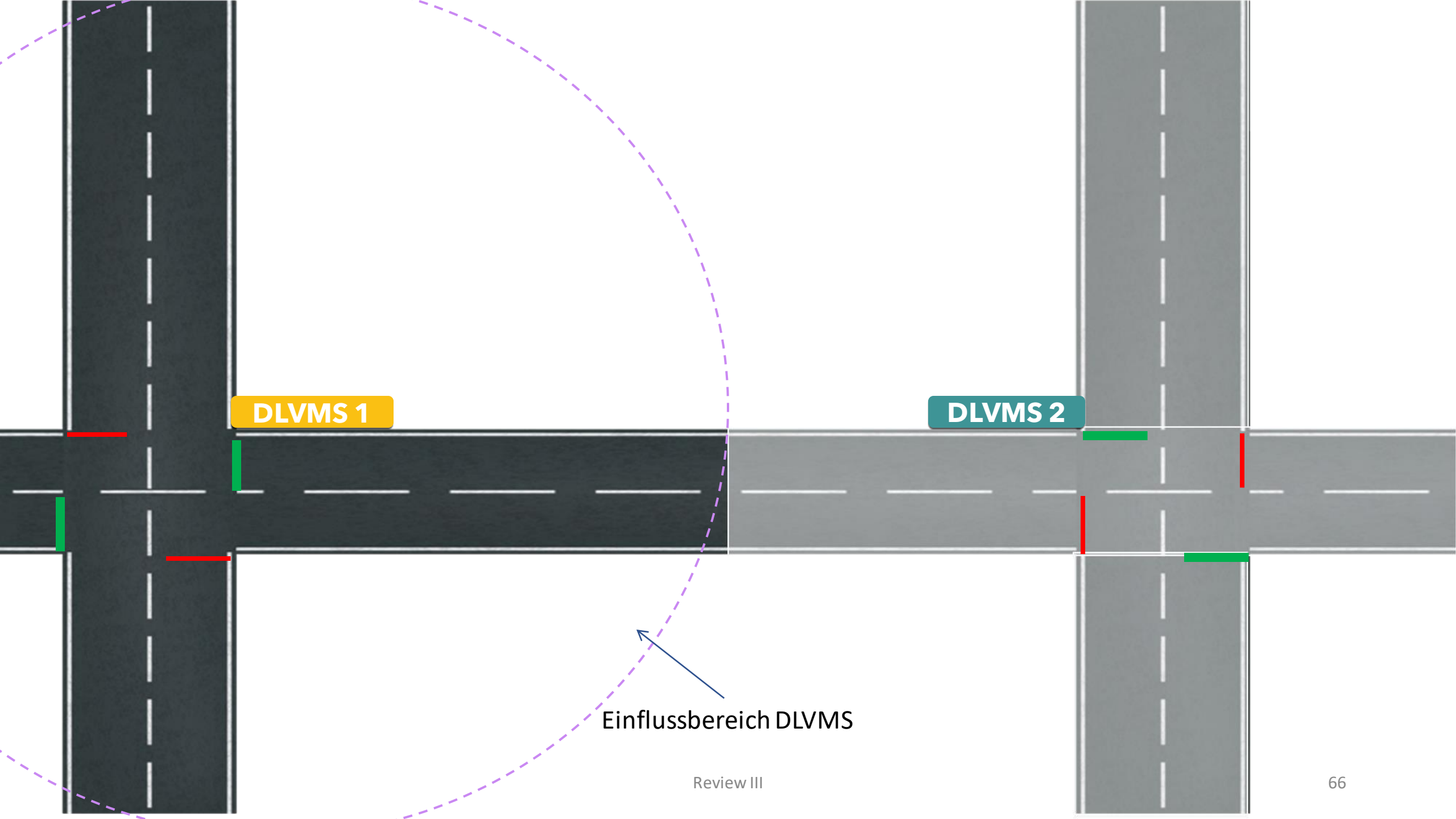
Agenda

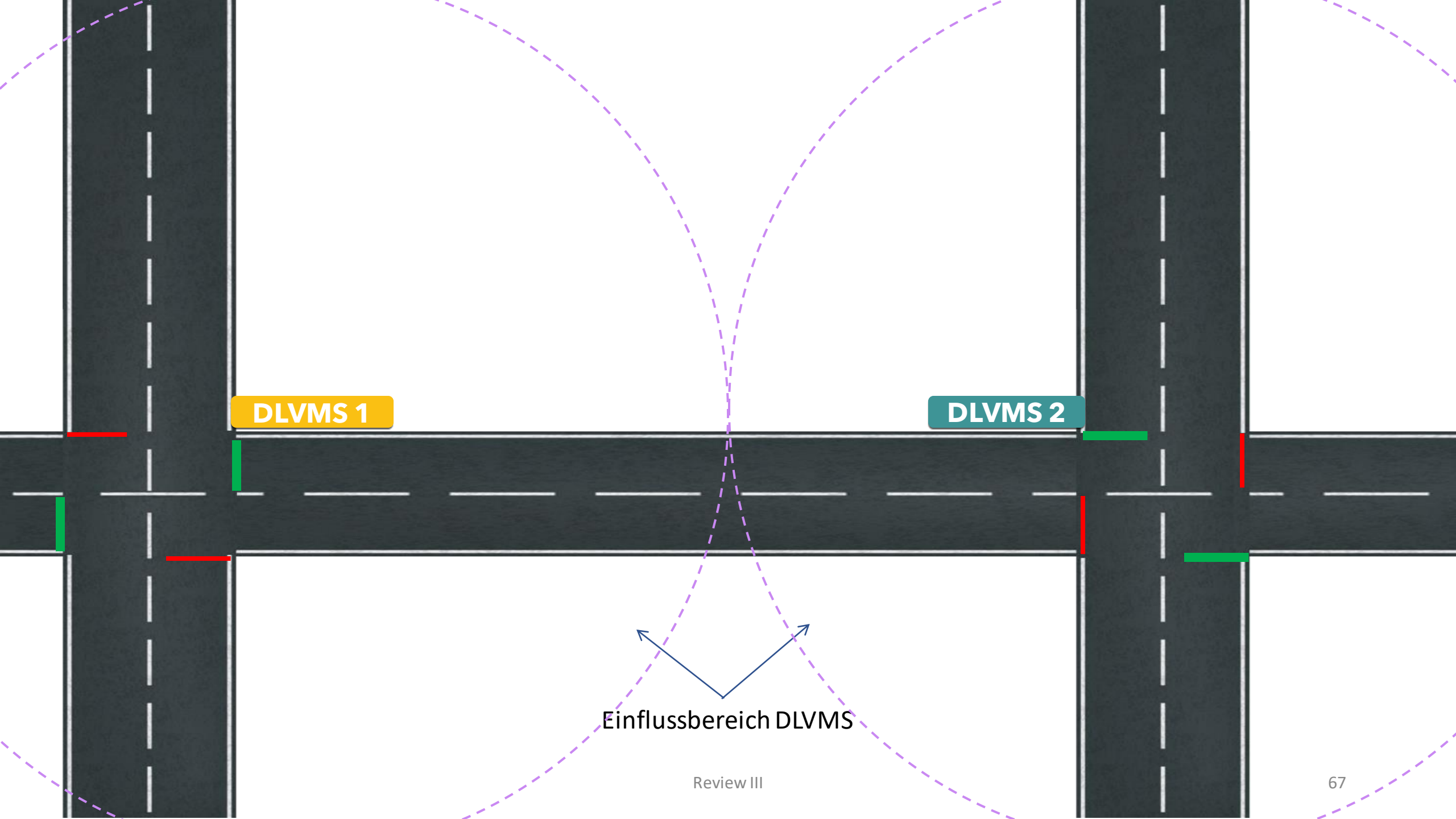
- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
  - Referenzsystem
  - Kaskadierendes Verfahren
  - LSA
  - Dynamische Rettungsgasse
  - KI-Konzepte
- Implementierung
- Test
- Demonstrator
- Evaluation
- Ausblick
- Fazit

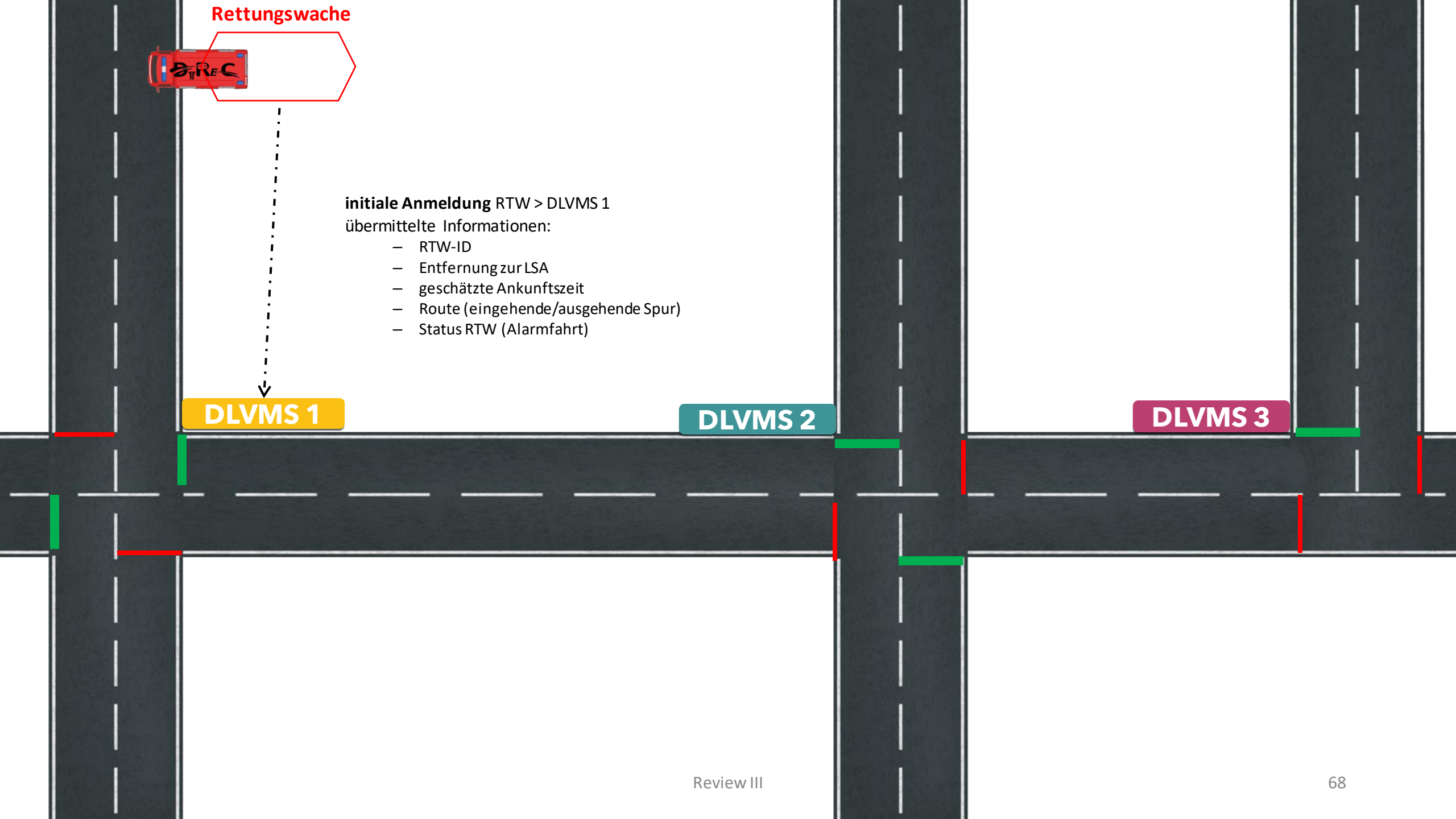

Referenzsystem
Ziel
Evaluation der realisierten Konzepte
Realität
112
ELR
Bestimmung der notwendigen Rettungsfahrzeuge
Start der Alarmfahrt
Verschiedene Konflikte
Notfälle Kategorien
Feuerausbruch
Beinfraktur
Herzinfarkt
Chemieunfall

Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
  - Referenzsystem
  - Kaskadierendes Verfahren
  - LSA
  - Dynamische Rettungsgasse
  - KI-Konzepte
- Implementierung
- Test
- Demonstrator
- Evaluation
- Ausblick
- Fazit


DLVMS 1
DLVMS 2
Einflussbereich DLVMS


DLVMS 1
DLVMS 2
Einflussbereich DLVMS



initiale Anmeldung RTW > DLVMS 1

übermittelte Informationen:

- RTW-ID
- Entfernung zur LSA
- geschätzte Ankunftszeit
- Route (eingehende/ausgehende Spur)
- Status RTW (Alarmfahrt)

Rettungswache

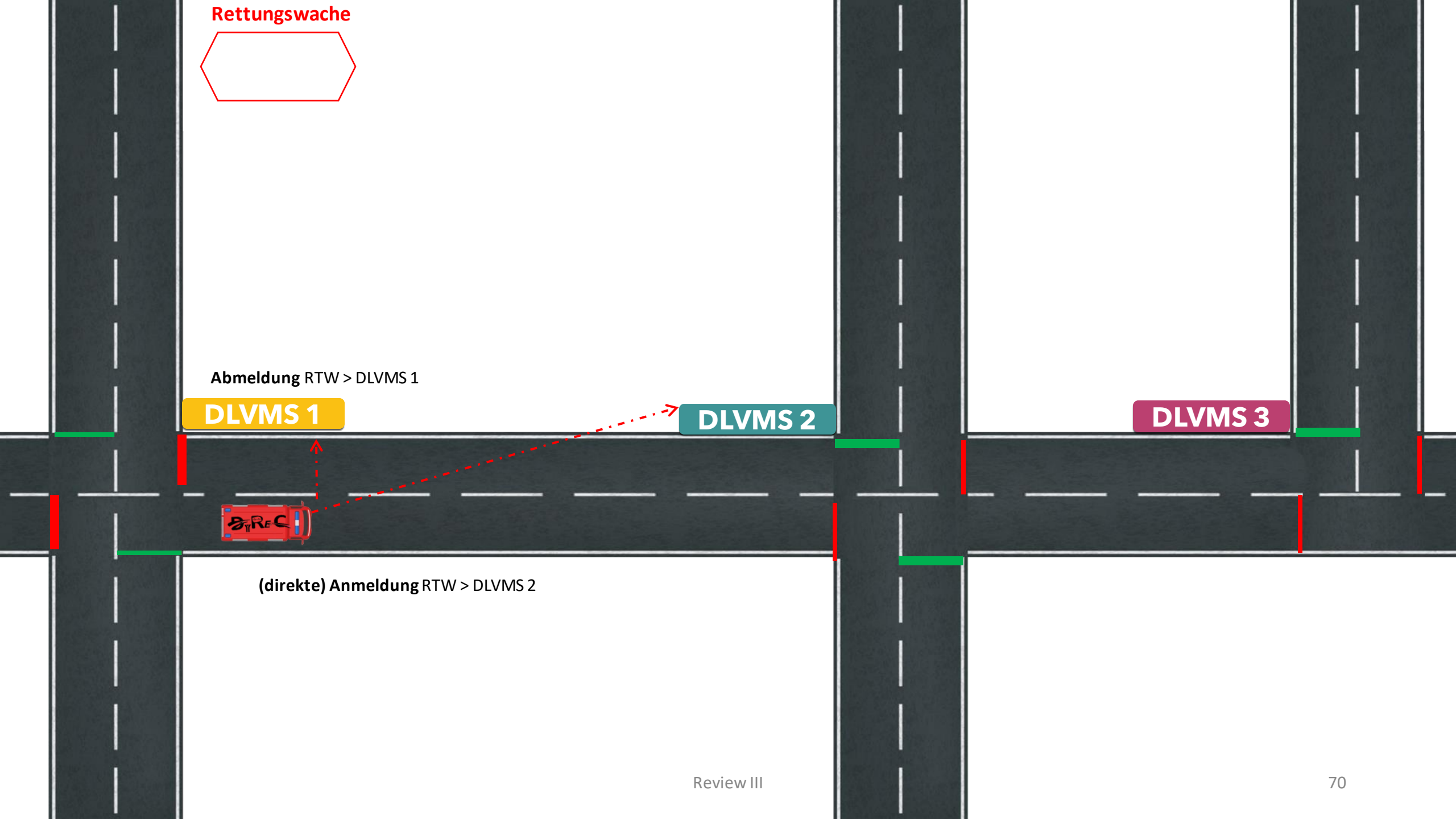
Vorankündigung/weiche Anmeldung DLVMS 1 > DLVMS 2
übermittelte Informationen:

- RTW-ID
- Entfernung zur LSA
- geschätzte Ankunftszeit
- Route (eingehende/ausgehende Spur)
- Status RTW (Alarmfahrt)
- Zuordnung DLVMS-1

DLVMS 1

DLVMS 2

DLVMS 3


Rettungswache
Abmeldung RTW > DLVMS 1
DLVMS 1
DLVMS 2
DLVMS 3
(direkte) Anmeldung RTW > DLVMS 2

Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
  - Referenzsystem
  - Kaskadierendes Verfahren
  - LSA
  - Dynamische Rettungsgasse
  - KI-Konzept
- Implementierung
- Test
- Demonstrator
- Evaluation
- Ausblick
- Fazit

- **roter Pfeil = Strecke**
- **Die Farbblöcke beschreiben die vom DLVMS 1 gesteuerten Ampelanlage**

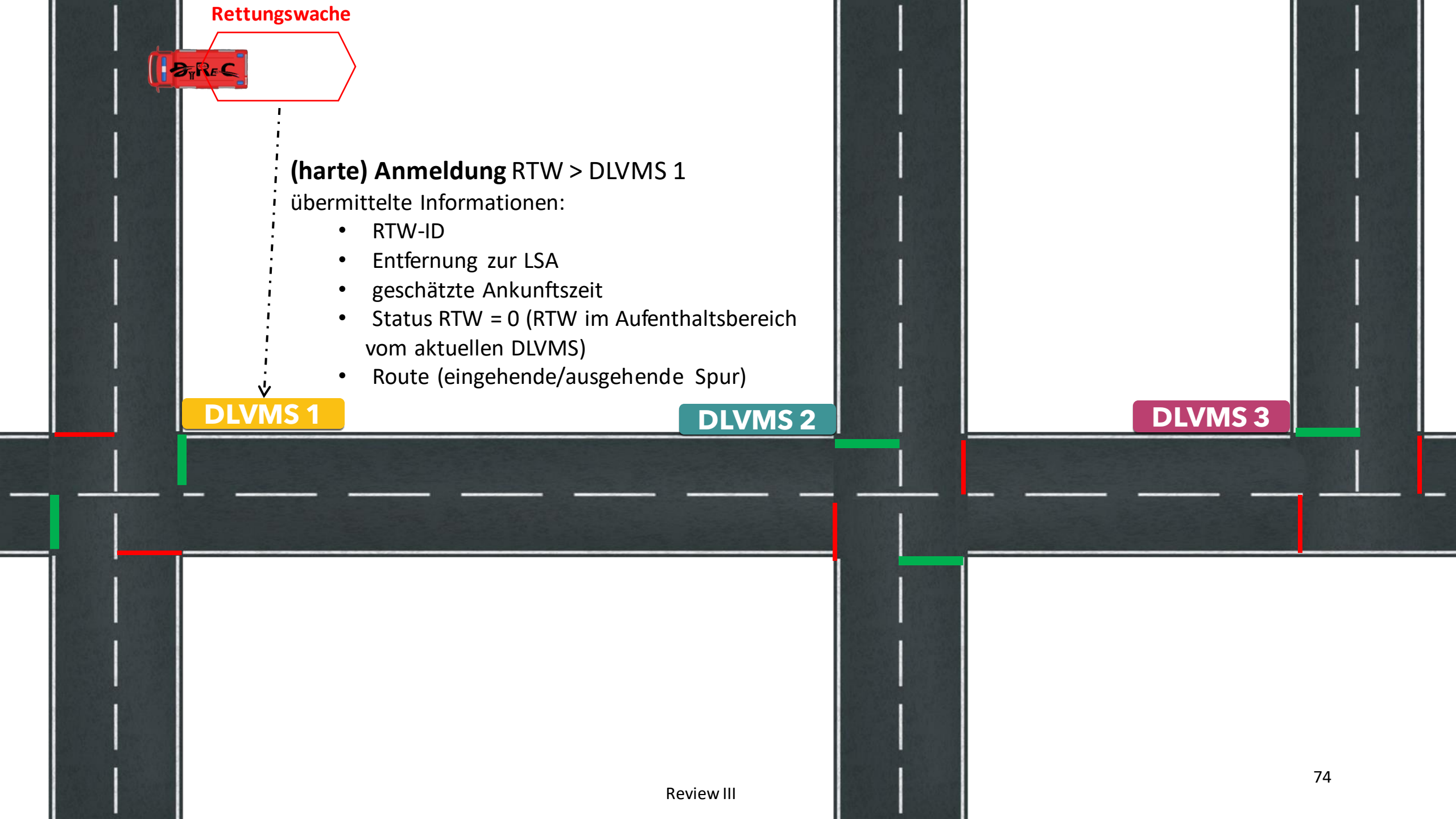
Rettungswache

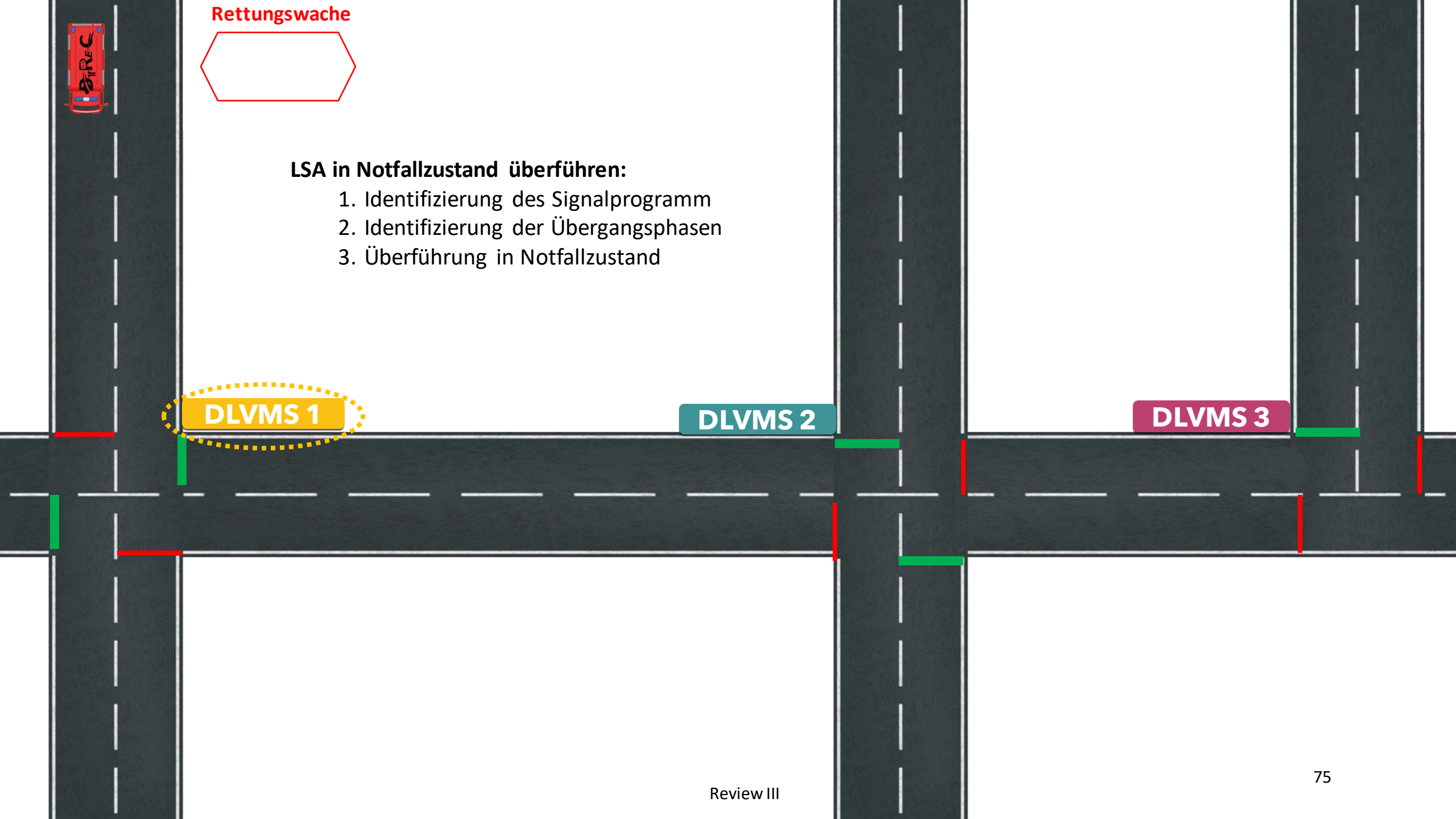
Aufenthaltsbereich der DLVMS

DLVMS 1

DLVMS 2

DLVMS 3


Rettungswache
(harte) Anmeldung RTW > DLVMS 1
übermittelte Informationen:
• RTW-ID
• Entfernung zur LSA
• geschätzte Ankunftszeit
• Status RTW = 0 (RTW im Aufenthaltsbereich vom aktuellen DLVMS)
• Route (eingehende/ausgehende Spur)
DLVMS 1
DLVMS 2
DLVMS 3


Rettungswache
LSA in Notfallzustand überführen:
1. Identifizierung des Signalprogramm
2. Identifizierung der Übergangsphasen
3. Überführung in Notfallzustand
DLVMS 1
DLVMS 2
DLVMS 3

Rettungswache

LSA in Notfallzustand überführen:

1. Identifizierung des Signalprogramm
2. Identifizierung der Übergangsphasen
3. Überführung in Notfallzustand

DLVMS 1

DLVMS 2

DLVMS 3

Rettungswache

LSA in Notfallzustand überführen:

1. Identifizierung des Signalprogramm
2. Identifizierung der Übergangsphasen
3. Überführung in Notfallzustand

DLVMS 1

DLVMS 2

DLVMS 3

Rettungswache

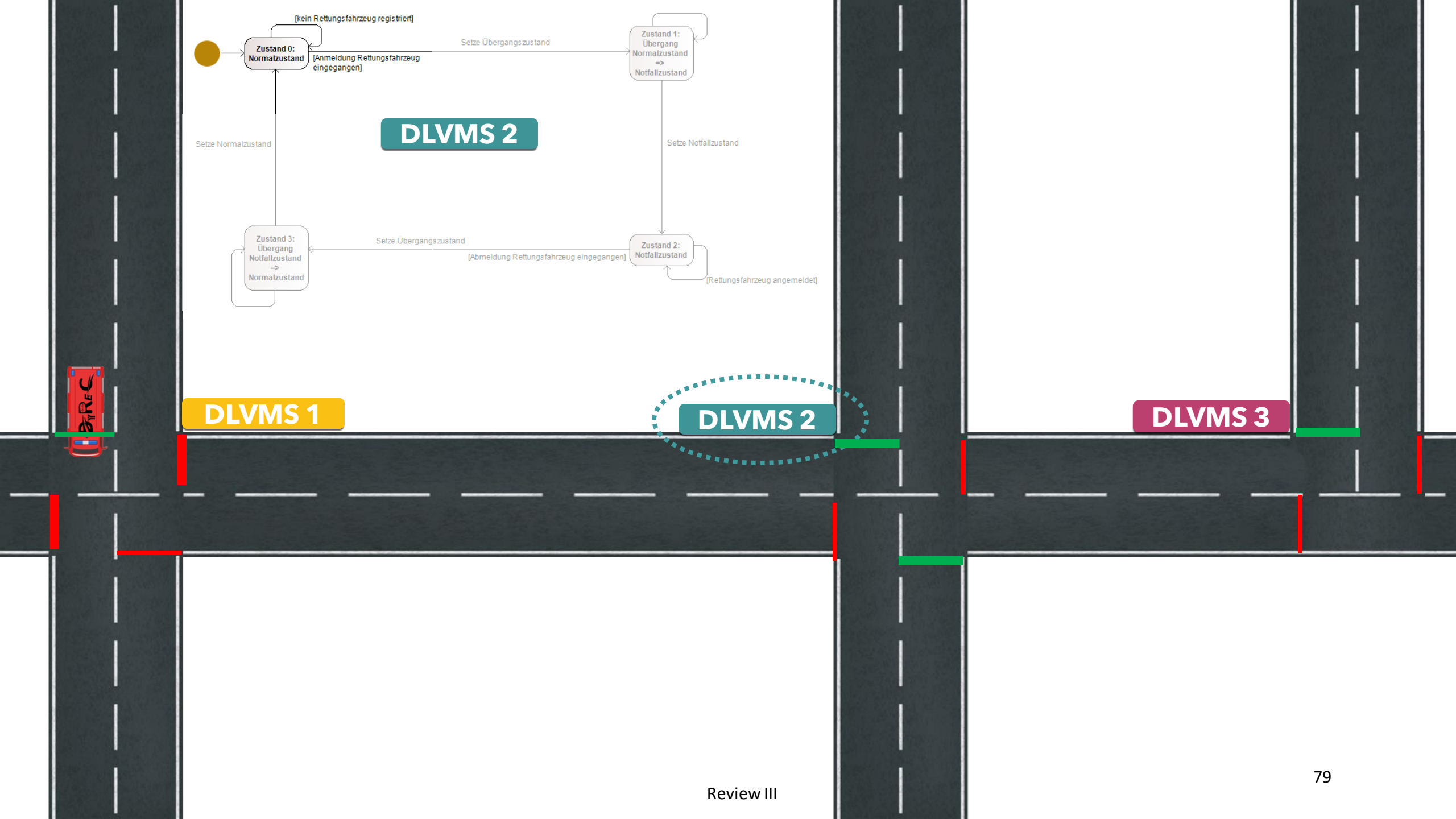
LSA in Notfallzustand überführen:

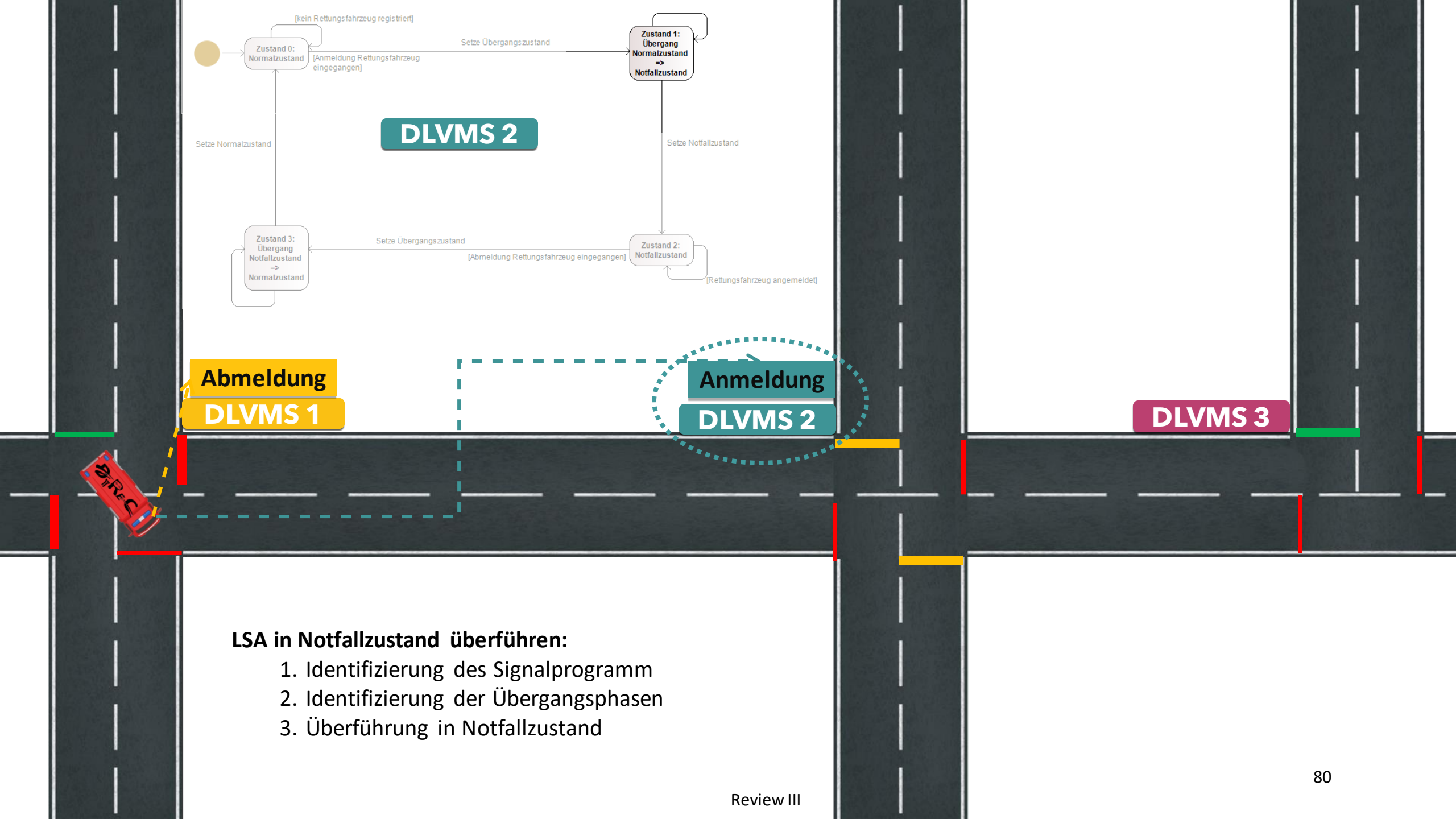
1. Identifizierung des Signalprogramm
2. Identifizierung der Übergangsphasen
3. Überführung in Notfallzustand

DLVMS 1

DLVMS 2

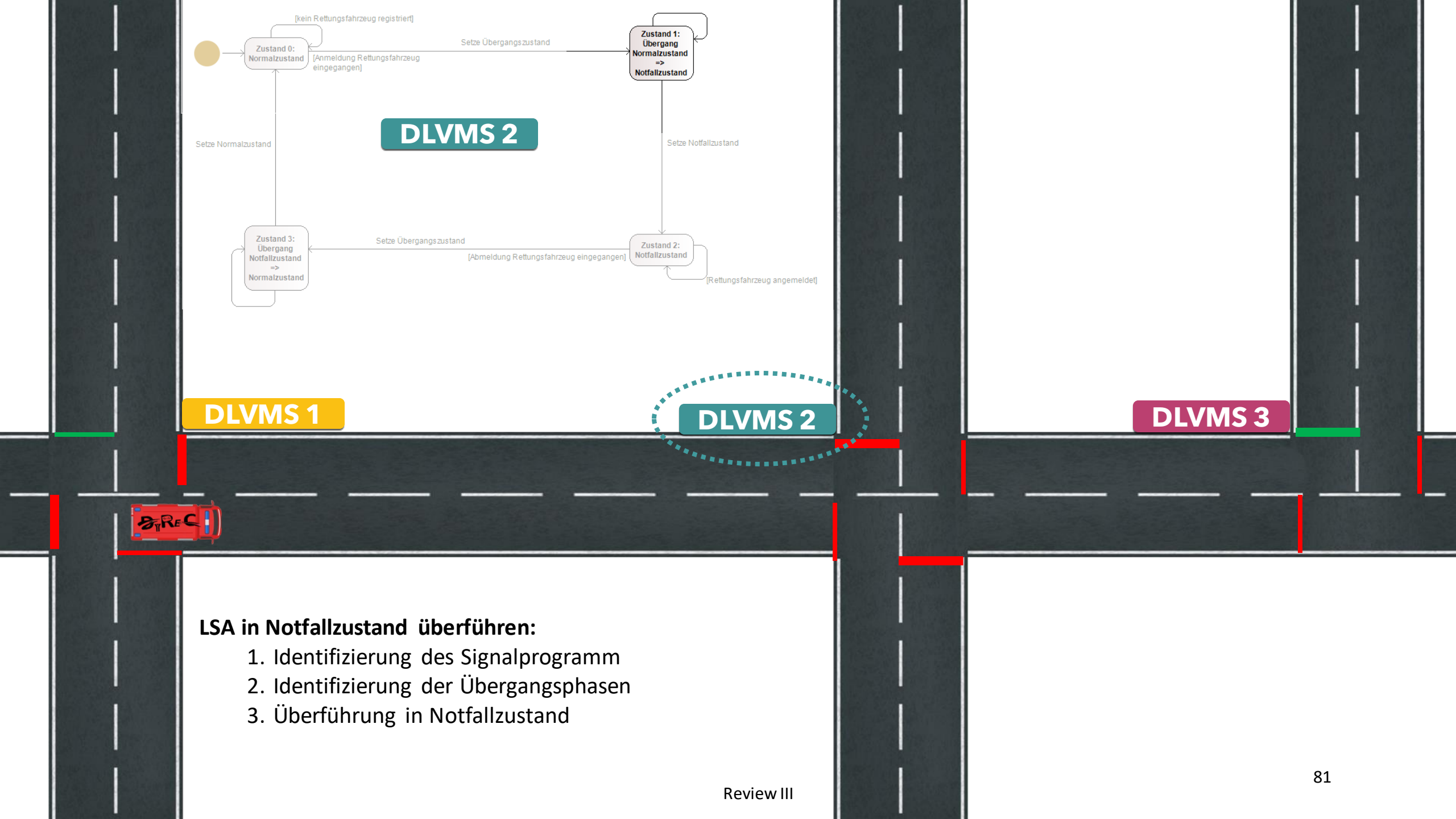
DLVMS 3


[kein Rettungsfahrzeug registriert]
Zustand 0: Normalzustand
[Anmeldung Rettungsfahrzeug eingegangen]
Setze Übergangszustand
Zustand 1: Übergang Normalzustand => Notfallzustand
Setze Notfallzustand
Zustand 2: Notfallzustand
[Rettungsfahrzeug angemeldet]
Setze Übergangszustand
[Abmeldung Rettungsfahrzeug eingegangen]
Zustand 3: Übergang Notfallzustand => Normalzustand
Setze Normalzustand
DLVMS 2
DLVMS 1
DLVMS 2
DLVMS 3



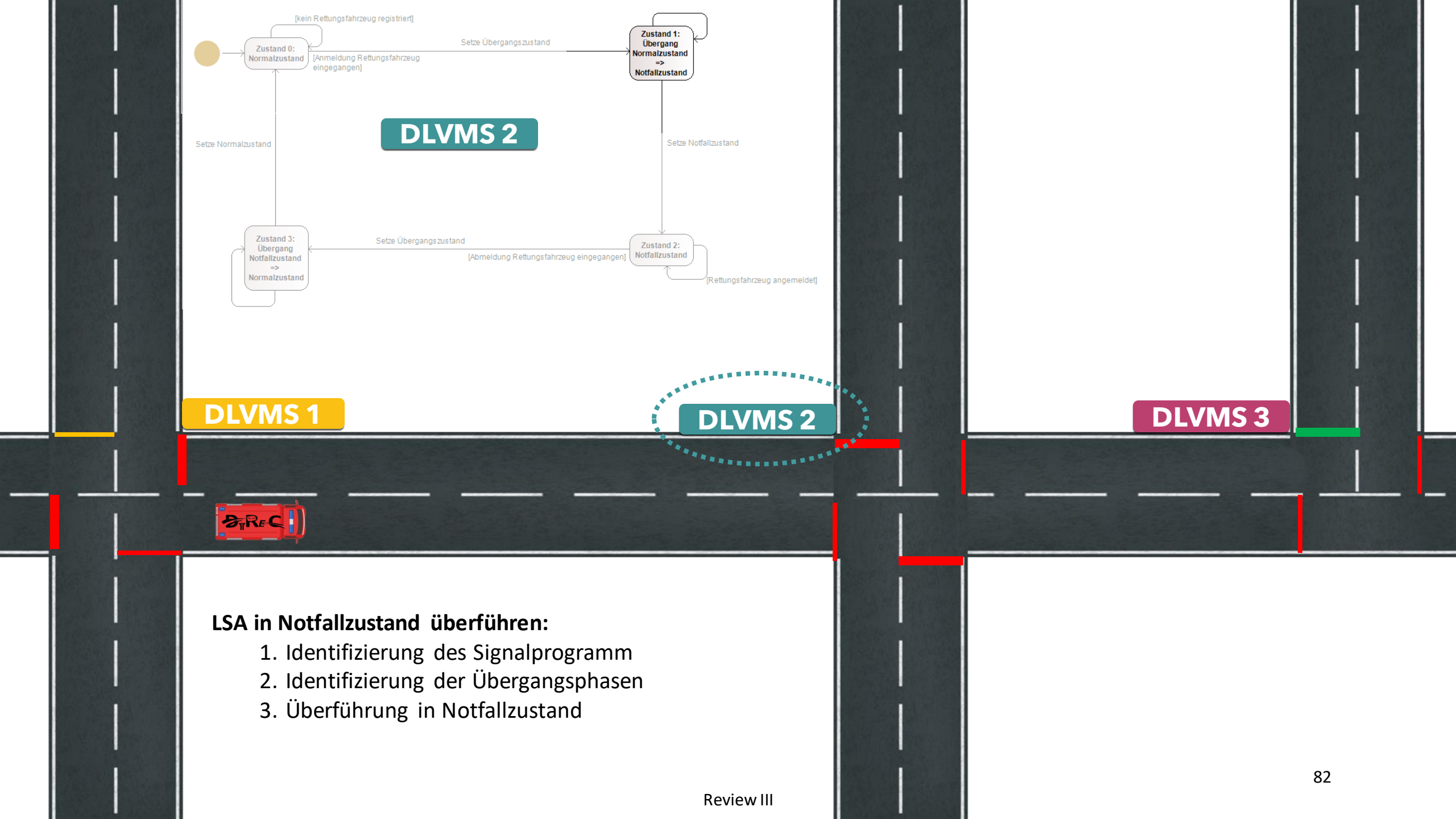
LSA in Notfallzustand überführen:

1. Identifizierung des Signalprogramm
2. Identifizierung der Übergangsphasen
3. Überführung in Notfallzustand



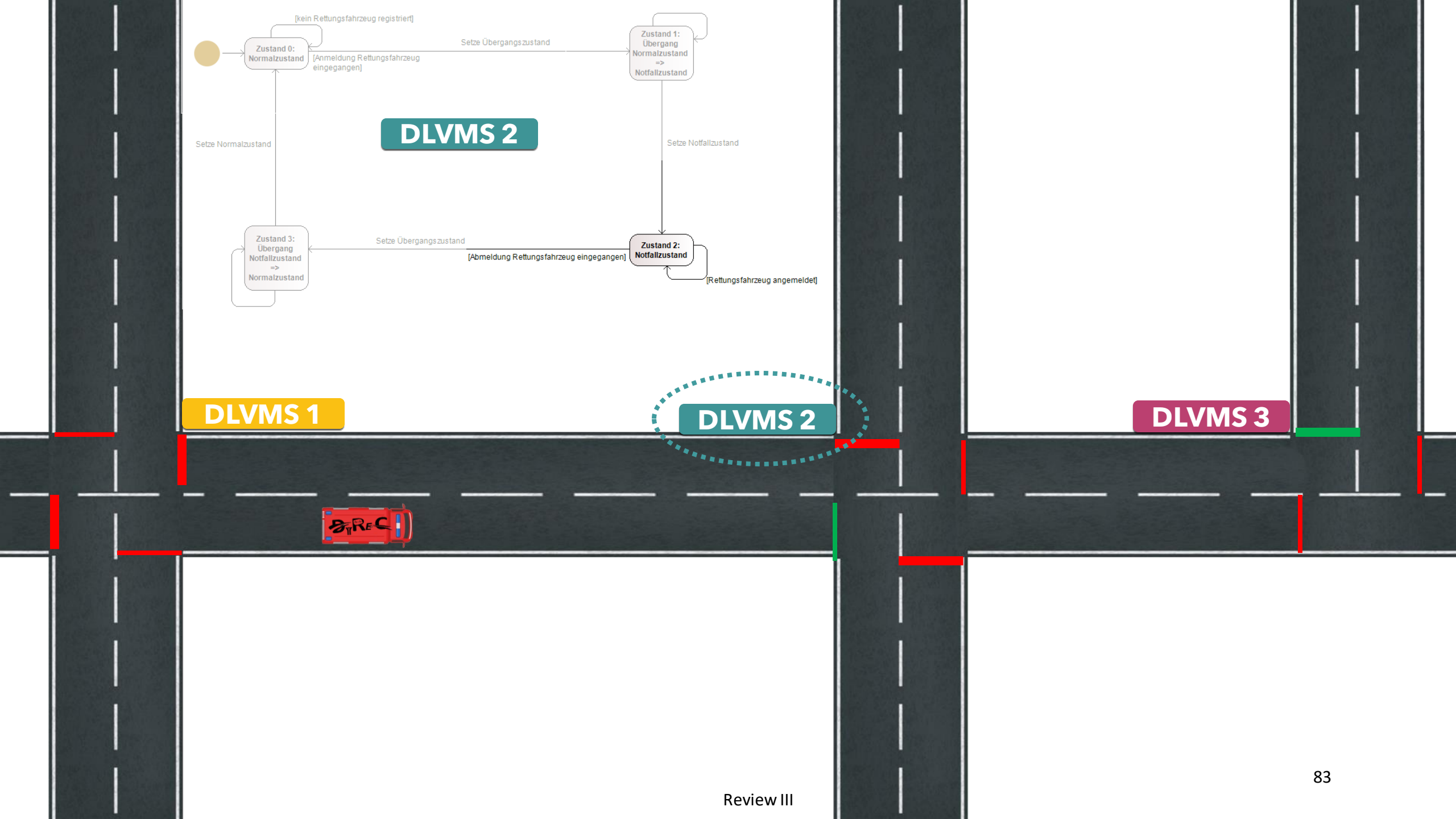
LSA in Notfallzustand überführen:

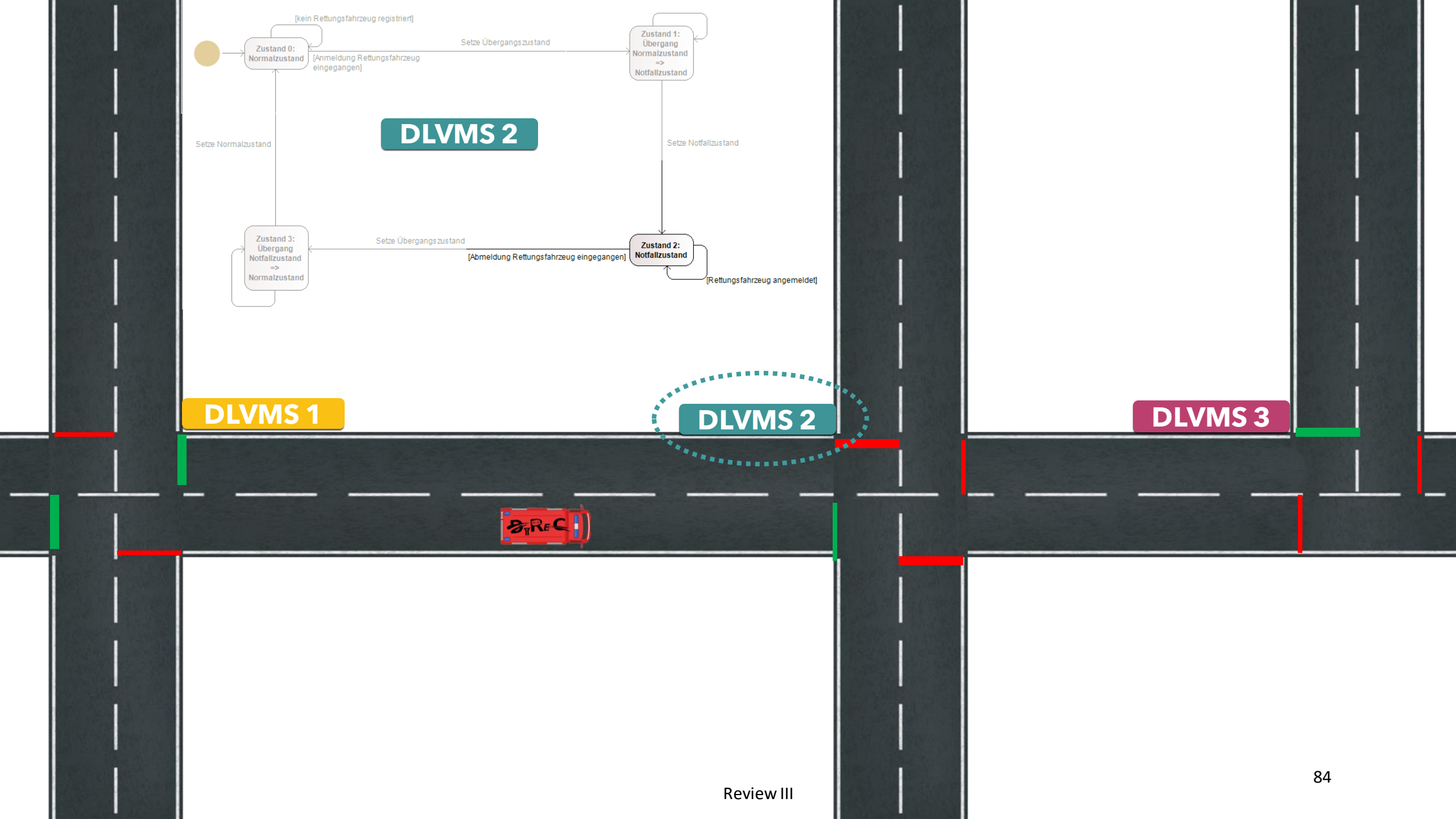
1. Identifizierung des Signalprogramm
2. Identifizierung der Übergangsphasen
3. Überführung in Notfallzustand

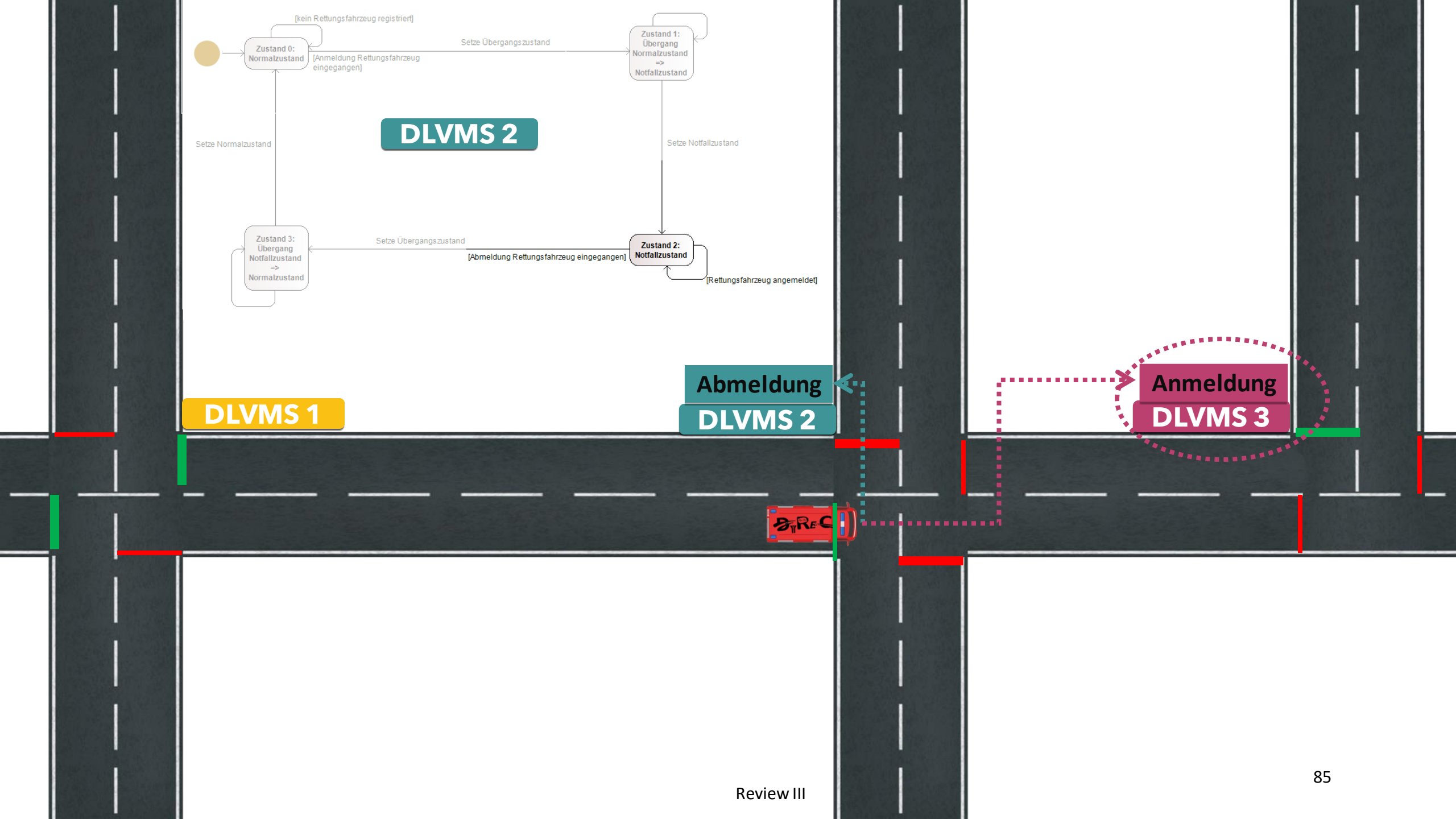


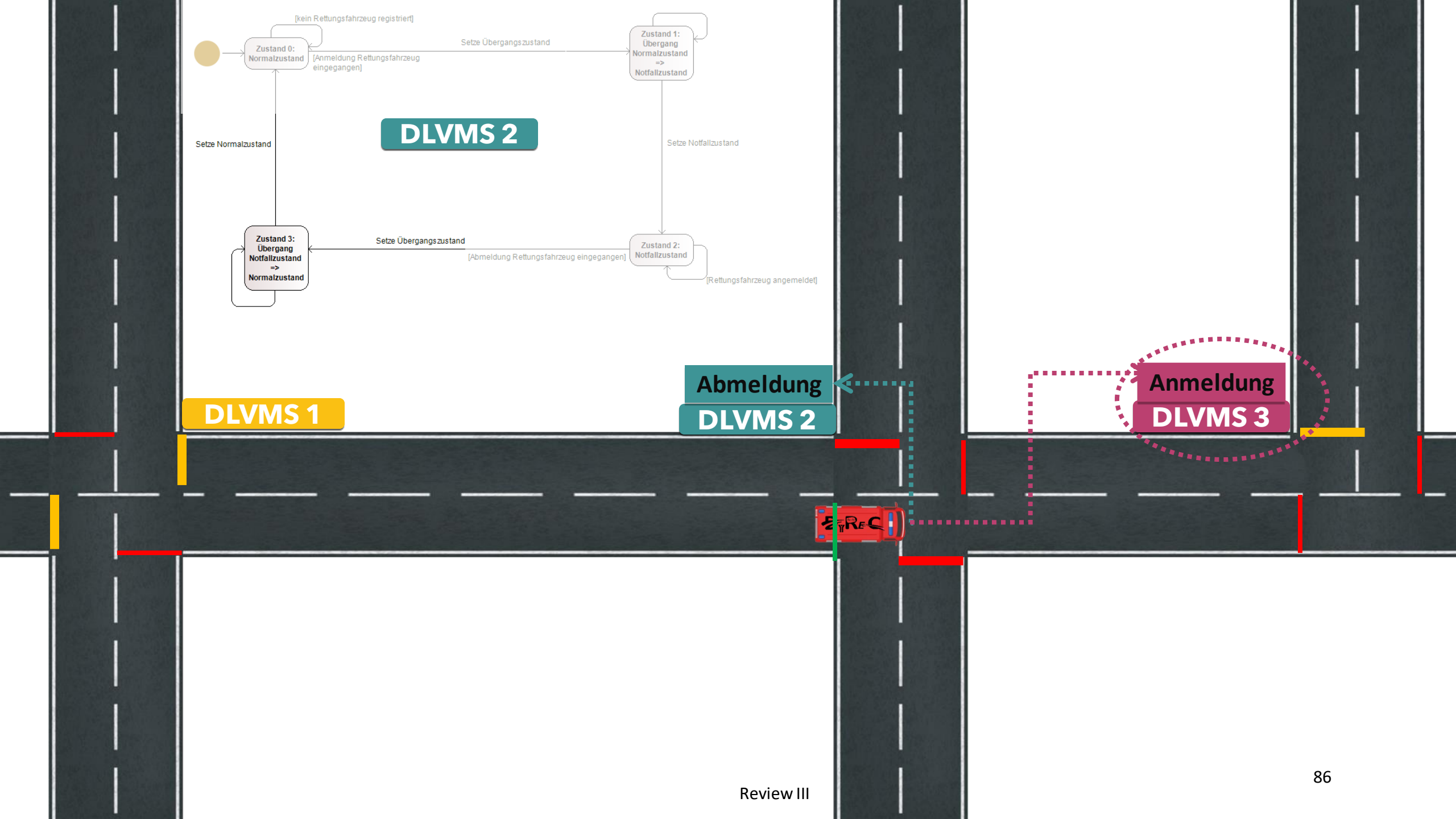
LSA in Notfallzustand überführen:

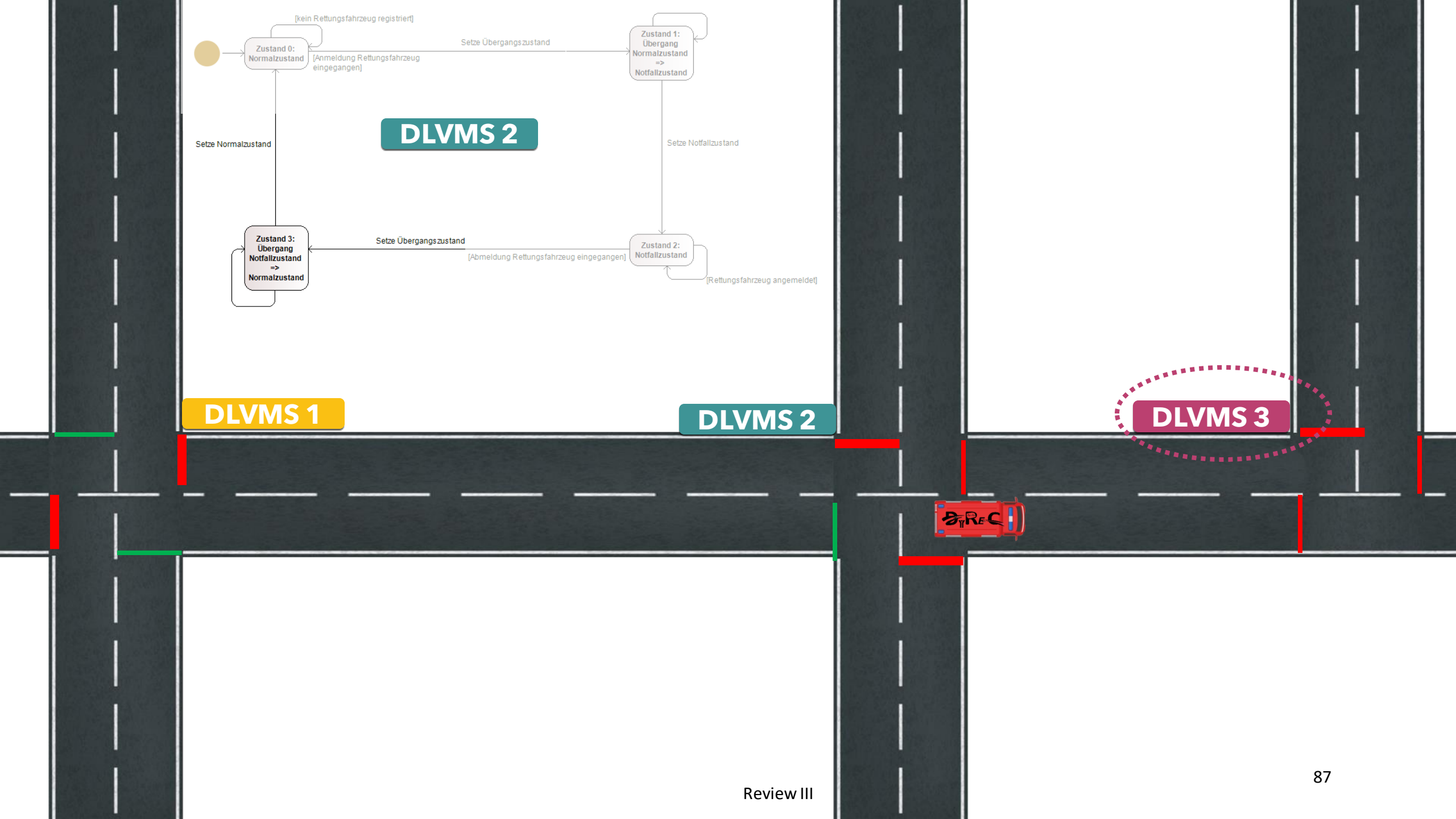
1. Identifizierung des Signalprogramm
2. Identifizierung der Übergangsphasen
3. Überführung in Notfallzustand

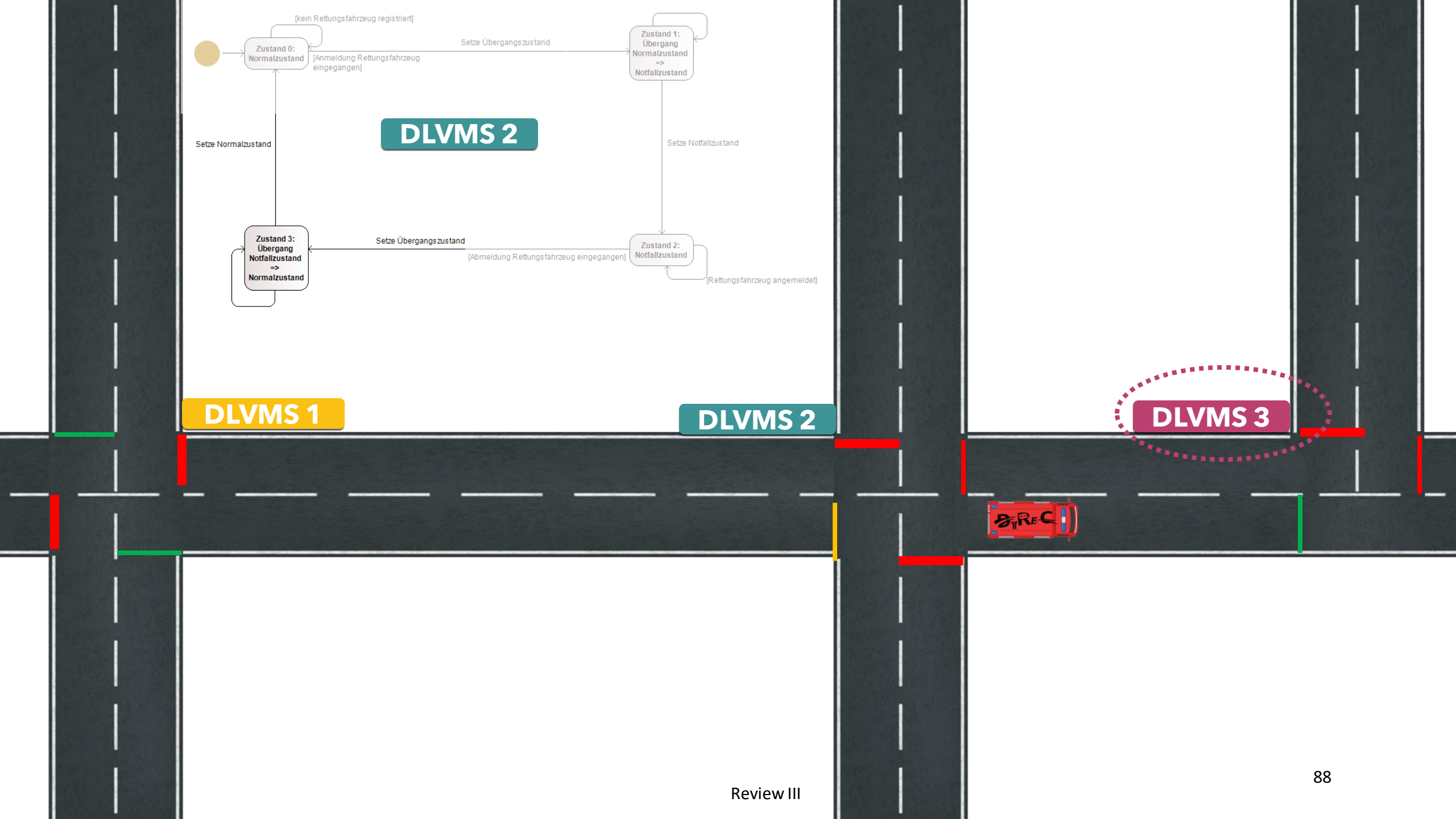

[kein Rettungsfahrzeug registriert]
Zustand 0: Normalzustand
Setze Übergangszustand
[Anmeldung Rettungsfahrzeug eingegangen]
Zustand 1: Übergang Normalzustand => Notfallzustand
DLVMS 2
Setze Normalzustand
Setze Notfallzustand
Zustand 3: Übergang Notfallzustand => Normalzustand
Setze Übergangszustand
[Abmeldung Rettungsfahrzeug eingegangen]
Zustand 2: Notfallzustand
[Rettungsfahrzeug angemeldet]
DLVMS 1
DLVMS 2
DLVMS 3

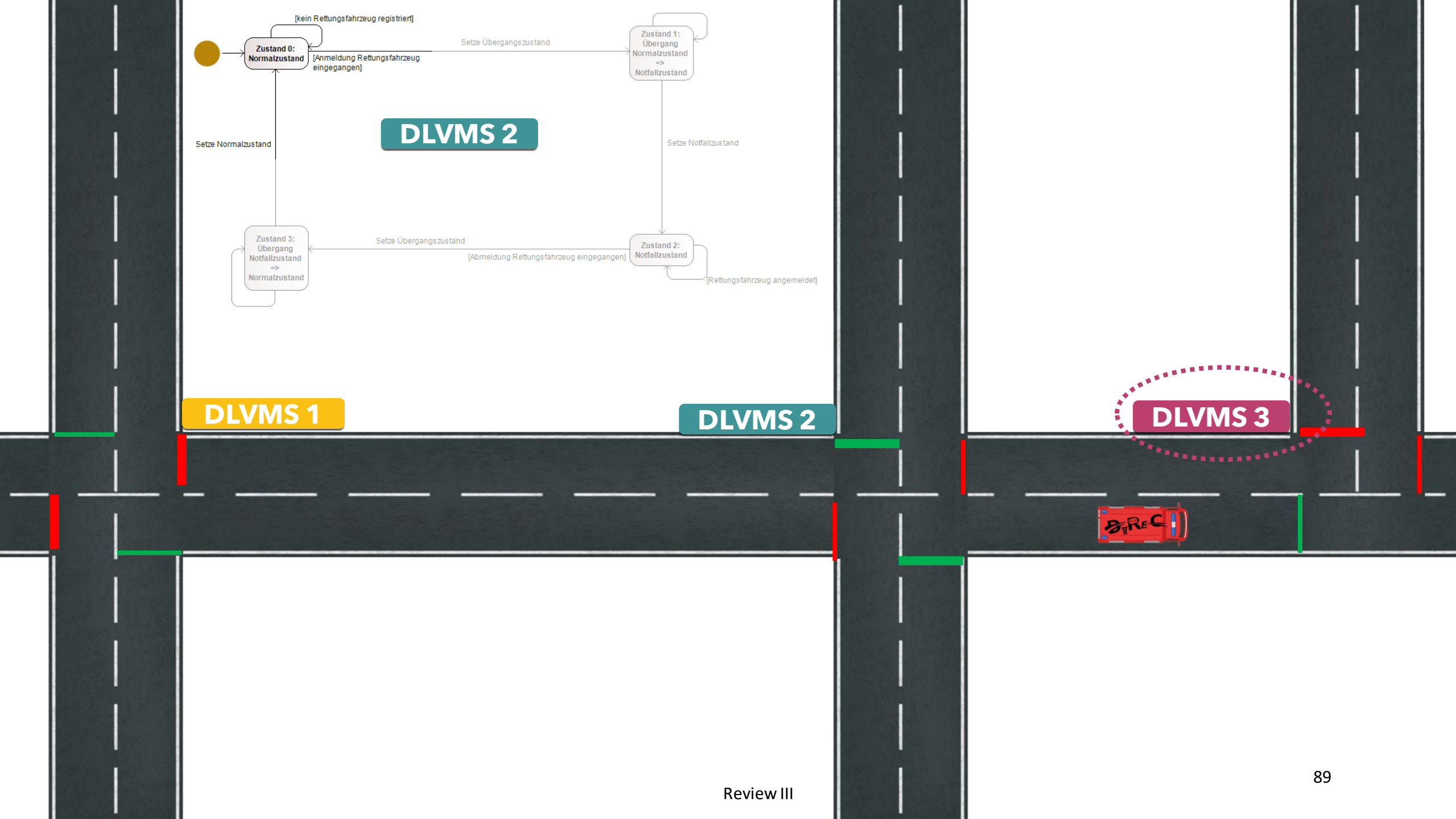

[kein Rettungsfahrzeug registriert]
Zustand 0: Normalzustand
Setze Übergangszustand
[Anmeldung Rettungsfahrzeug eingegangen]
Zustand 1: Übergang Normalzustand => Notfallzustand
DLVMS 2
Setze Normalzustand
Setze Notfallzustand
Zustand 3: Übergang Notfallzustand => Normalzustand
Setze Übergangszustand
[Abmeldung Rettungsfahrzeug eingegangen]
Zustand 2: Notfallzustand
[Rettungsfahrzeug angemeldet]
DLVMS 1
DLVMS 2
DLVMS 3

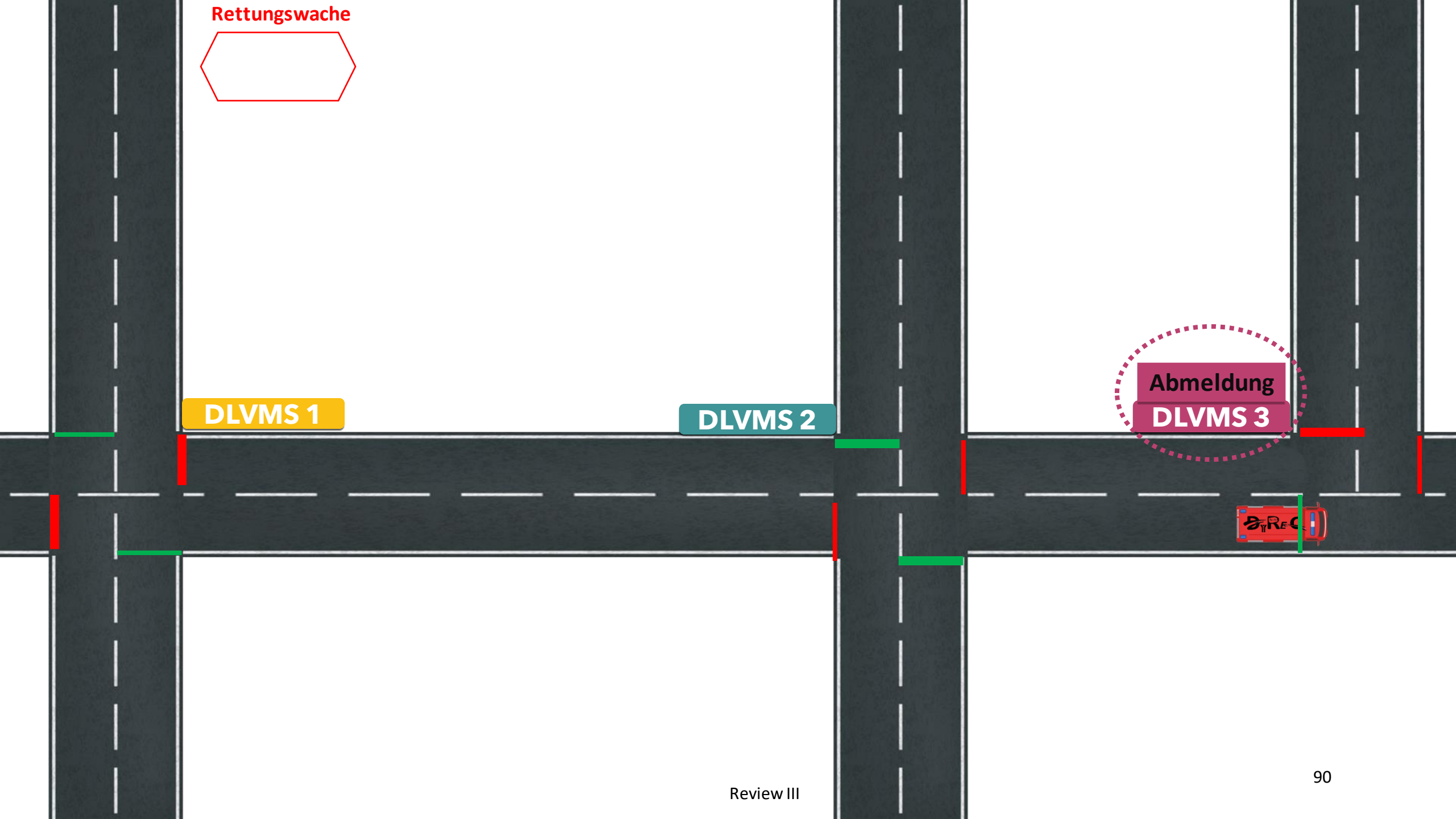

[kein Rettungsfahrzeug registriert]
Zustand 0: Normalzustand
Setze Übergangszustand
[Anmeldung Rettungsfahrzeug eingegangen]
Zustand 1: Übergang Normalzustand => Notfallzustand
DLVMS 2
Setze Normalzustand
Setze Notfallzustand
Zustand 3: Übergang Notfallzustand => Normalzustand
Setze Übergangszustand
[Abmeldung Rettungsfahrzeug eingegangen]
Zustand 2: Notfallzustand
[Rettungsfahrzeug angemeldet]
DLVMS 1
Abmeldung
DLVMS 2
Anmeldung
DLVMS 3

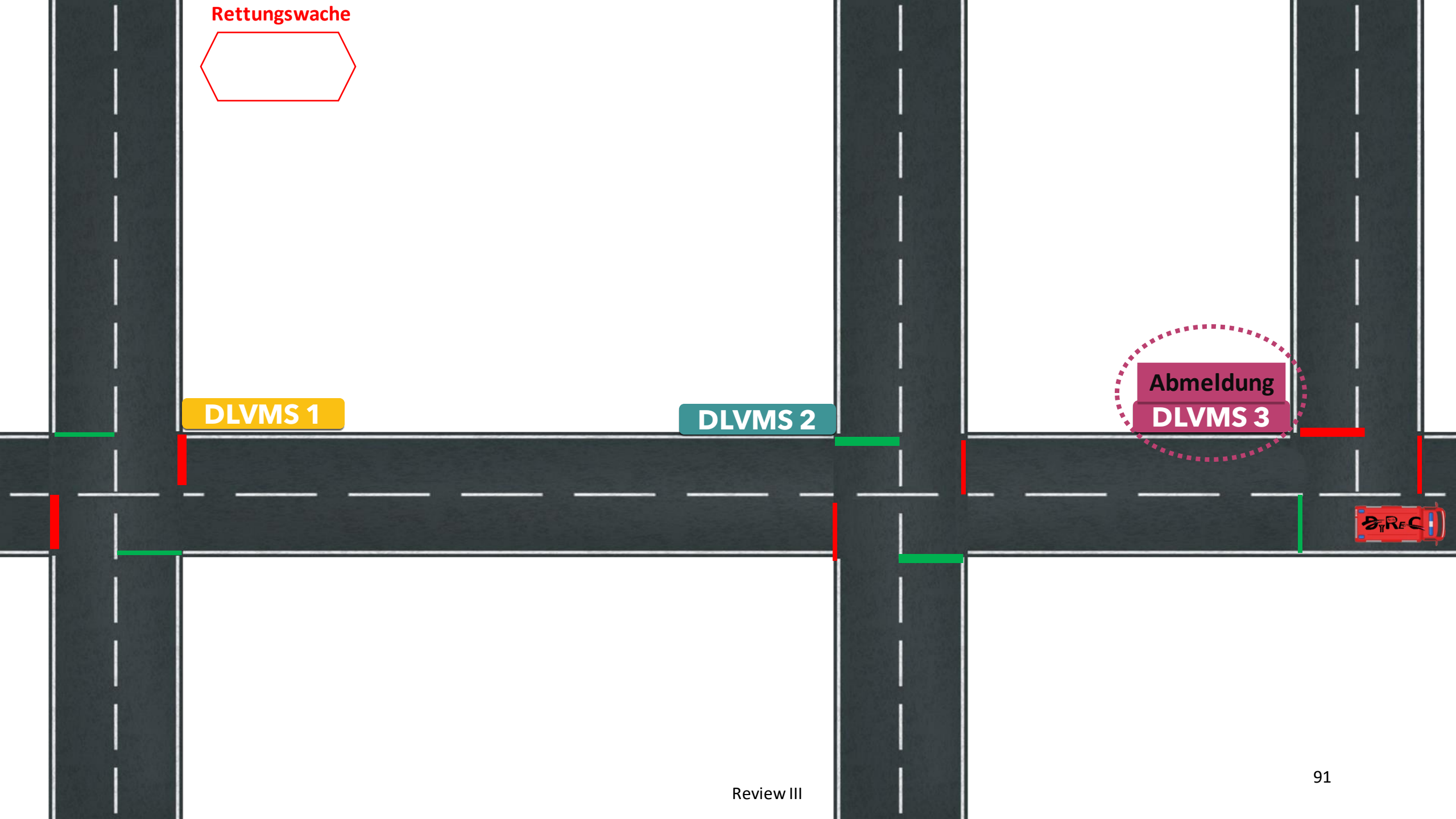

[kein Rettungsfahrzeug registriert]
Zustand 0: Normalzustand
Setze Übergangszustand
[Anmeldung Rettungsfahrzeug eingegangen]
Zustand 1: Übergang Normalzustand => Notfallzustand
DLVMS 2
Setze Normalzustand
Setze Notfallzustand
Zustand 3: Übergang Notfallzustand => Normalzustand
Setze Übergangszustand
[Abmeldung Rettungsfahrzeug eingegangen]
Zustand 2: Notfallzustand
[Rettungsfahrzeug angemeldet]
DLVMS 1
Abmeldung
DLVMS 2
Anmeldung
DLVMS 3


[kein Rettungsfahrzeug registriert]
Zustand 0: Normalzustand
Setze Übergangszustand
[Anmeldung Rettungsfahrzeug eingegangen]
Zustand 1: Übergang Normalzustand => Notfallzustand
DLVMS 2
Setze Normalzustand
Setze Notfallzustand
Zustand 3: Übergang Notfallzustand => Normalzustand
Setze Übergangszustand
[Abmeldung Rettungsfahrzeug eingegangen]
Zustand 2: Notfallzustand
[Rettungsfahrzeug angemeldet]
DLVMS 1
DLVMS 2
DLVMS 3


[kein Rettungsfahrzeug registriert]
Zustand 0: Normalzustand
Setze Übergangszustand
[Anmeldung Rettungsfahrzeug eingegangen]
Zustand 1: Übergang Normalzustand => Notfallzustand
DLVMS 2
Setze Normalzustand
Setze Notfallzustand
Zustand 3: Übergang Notfallzustand => Normalzustand
Setze Übergangszustand
[Abmeldung Rettungsfahrzeug eingegangen]
Zustand 2: Notfallzustand
[Rettungsfahrzeug angemeldet]
DLVMS 1
DLVMS 2
DLVMS 3


[kein Rettungsfahrzeug registriert]
Zustand 0: Normalzustand
[Anmeldung Rettungsfahrzeug eingegangen]
Setze Übergangszustand
Zustand 1: Übergang Normalzustand => Notfallzustand
Setze Normalzustand
Setze Notfallzustand
DLVMS 2
Zustand 3: Übergang Notfallzustand => Normalzustand
Setze Übergangszustand
[Abmeldung Rettungsfahrzeug eingegangen]
Zustand 2: Notfallzustand
[Rettungsfahrzeug angemeldet]
DLVMS 1
DLVMS 2
DLVMS 3


Rettungswache
DLVMS 1
DLVMS 2
Abmeldung
DLVMS 3


Rettungswache
DLVMS 1
DLVMS 2
Abmeldung
DLVMS 3

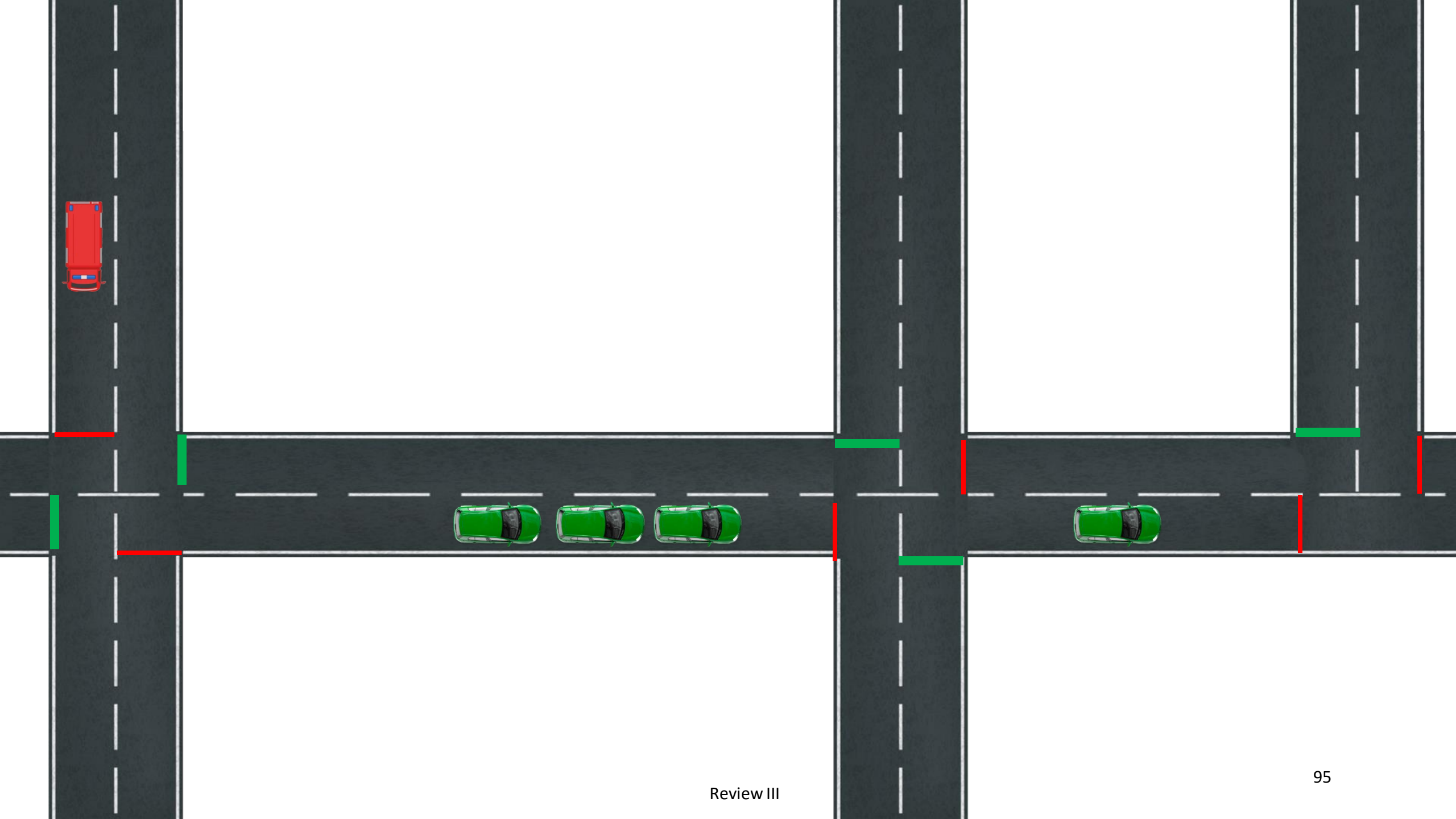
Agenda

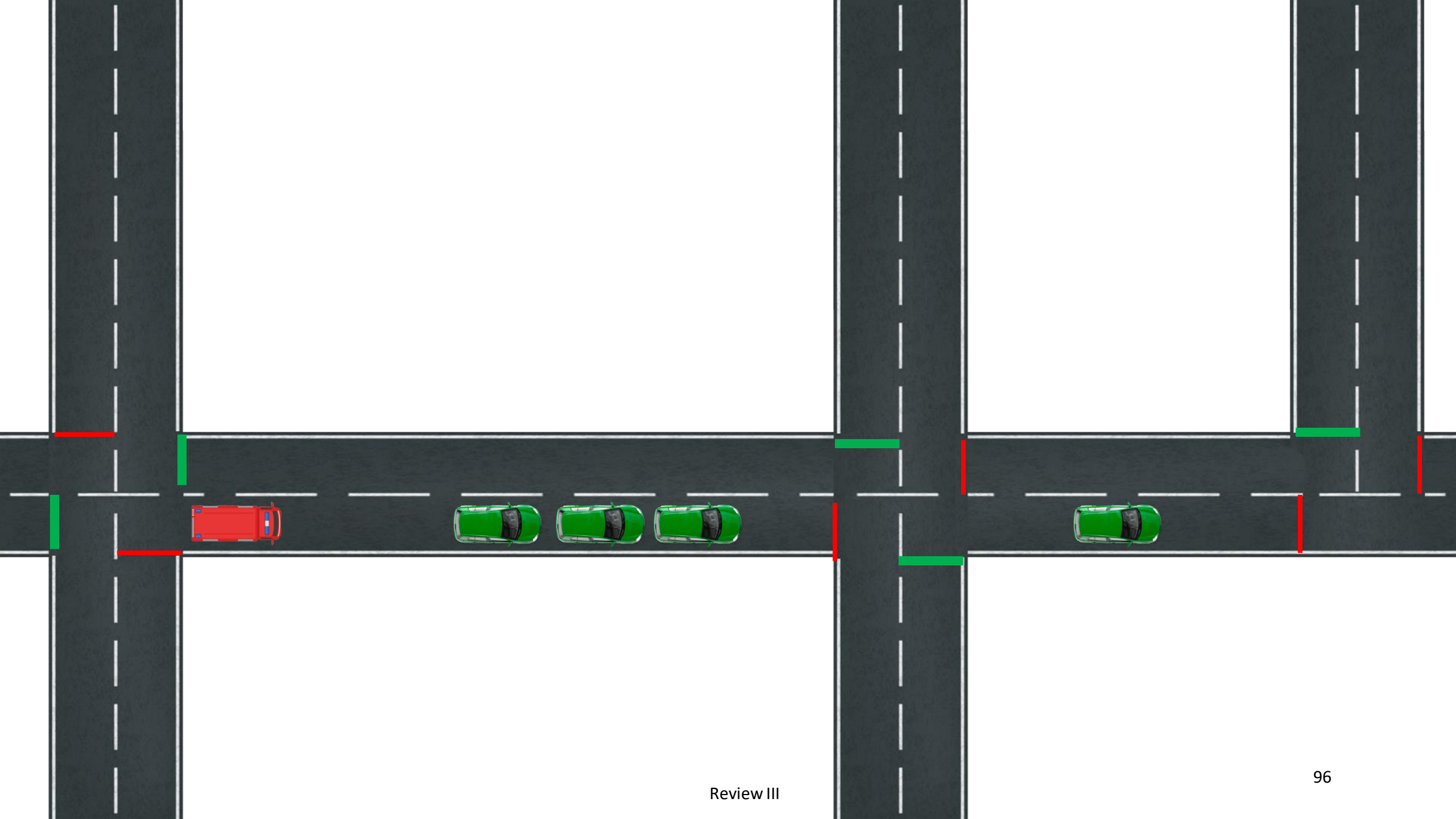
- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
  - Referenzsystem
  - Kaskadierendes Verfahren
  - LSA
  - Dynamische Rettungsgasse
  - KI-Konzept
- Implementierung
- Test
- Demonstrator
- Evaluation
- Ausblick
- Fazit

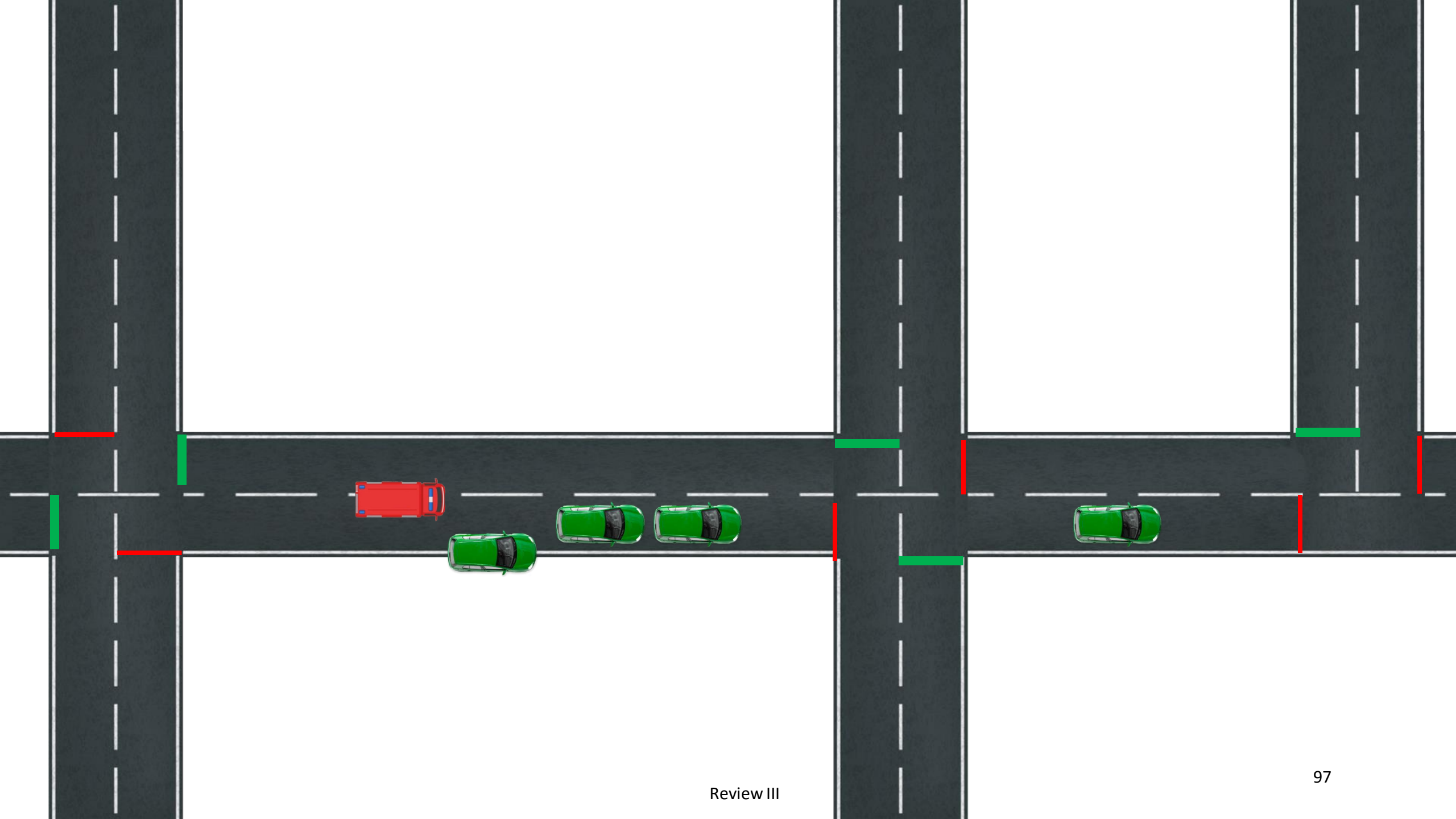
Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
  - Referenzsystem
  - Kaskadierendes Verfahren
  - LSA
  - Dynamische Rettungsgasse
  - KI-Konzept
- Implementierung
- Test
- Demonstrator
- Evaluation
- Ausblick
- Fazit

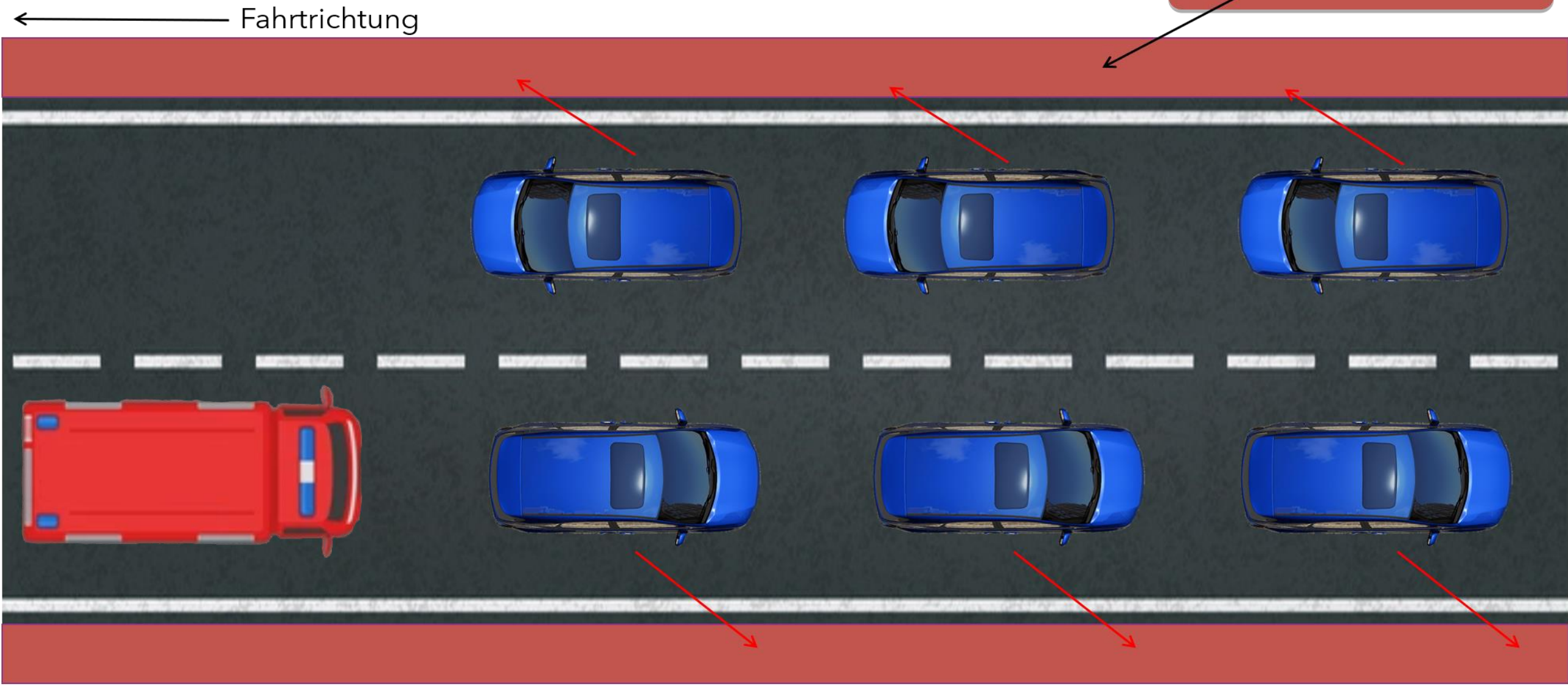
Rettungsgasse in der Realtiät - ohne Einsatz von DyReC



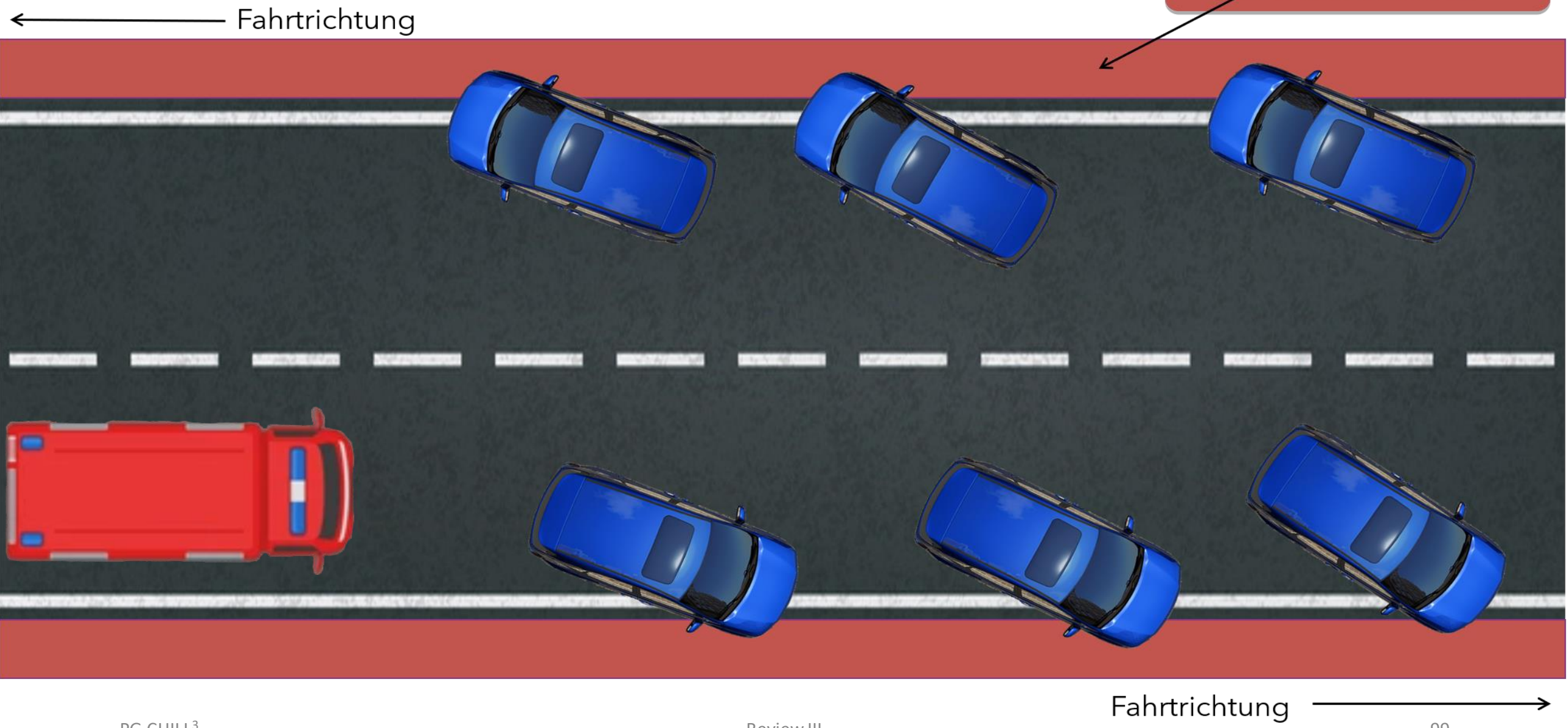




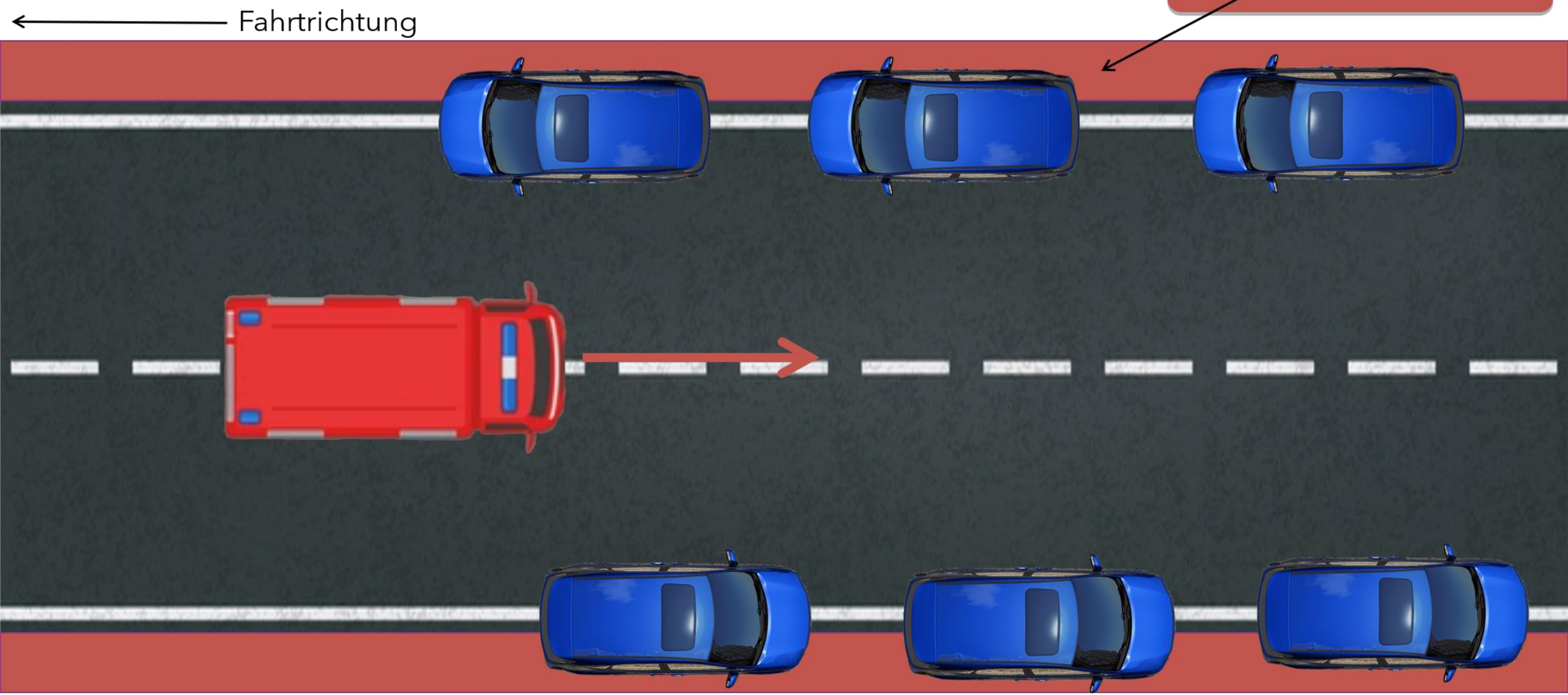
Dynamische Rettungsgasse in der Realität



Dynamische Rettungsgasse in der Realität

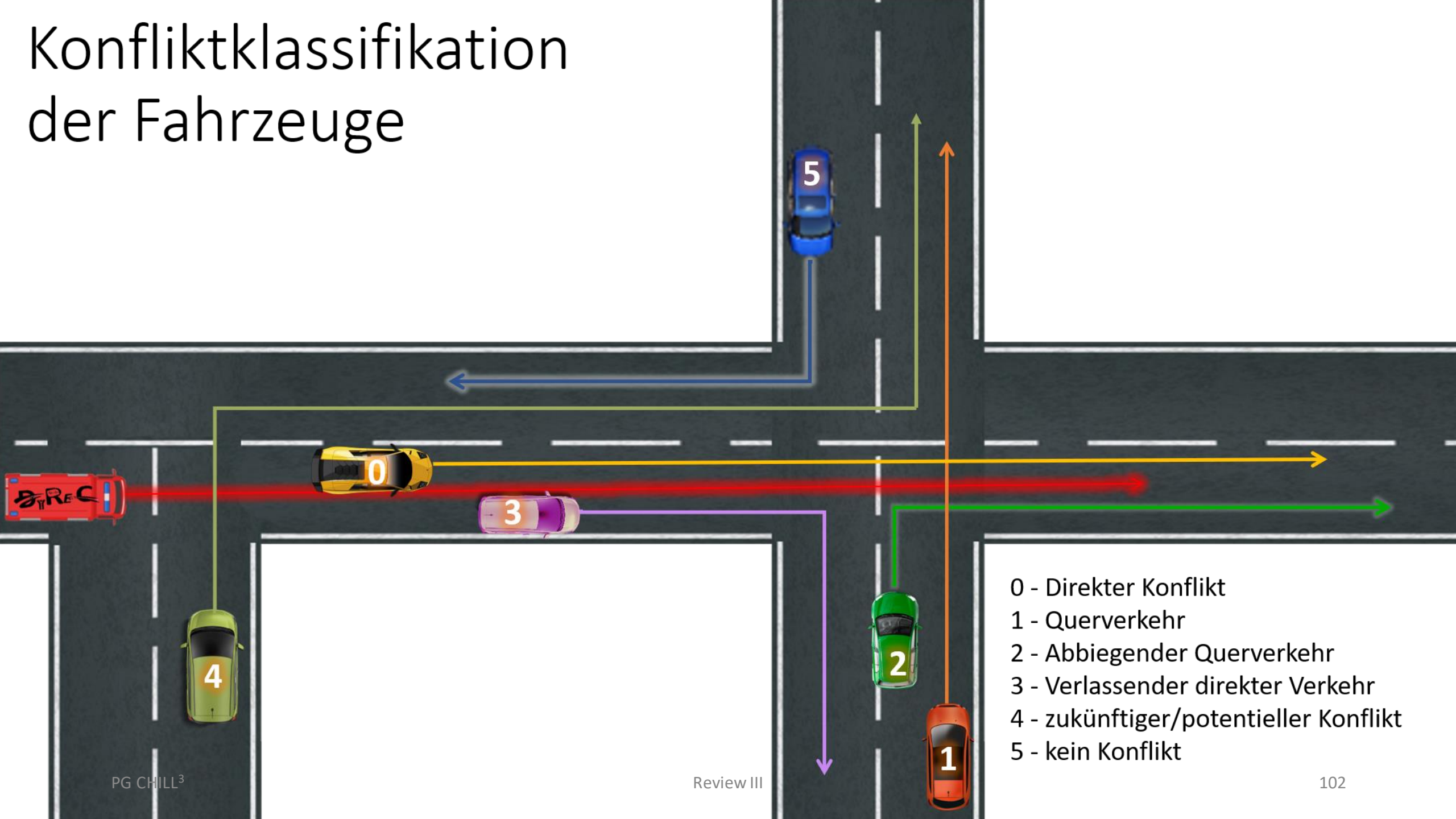


Dynamische Rettungsgasse in der Realität

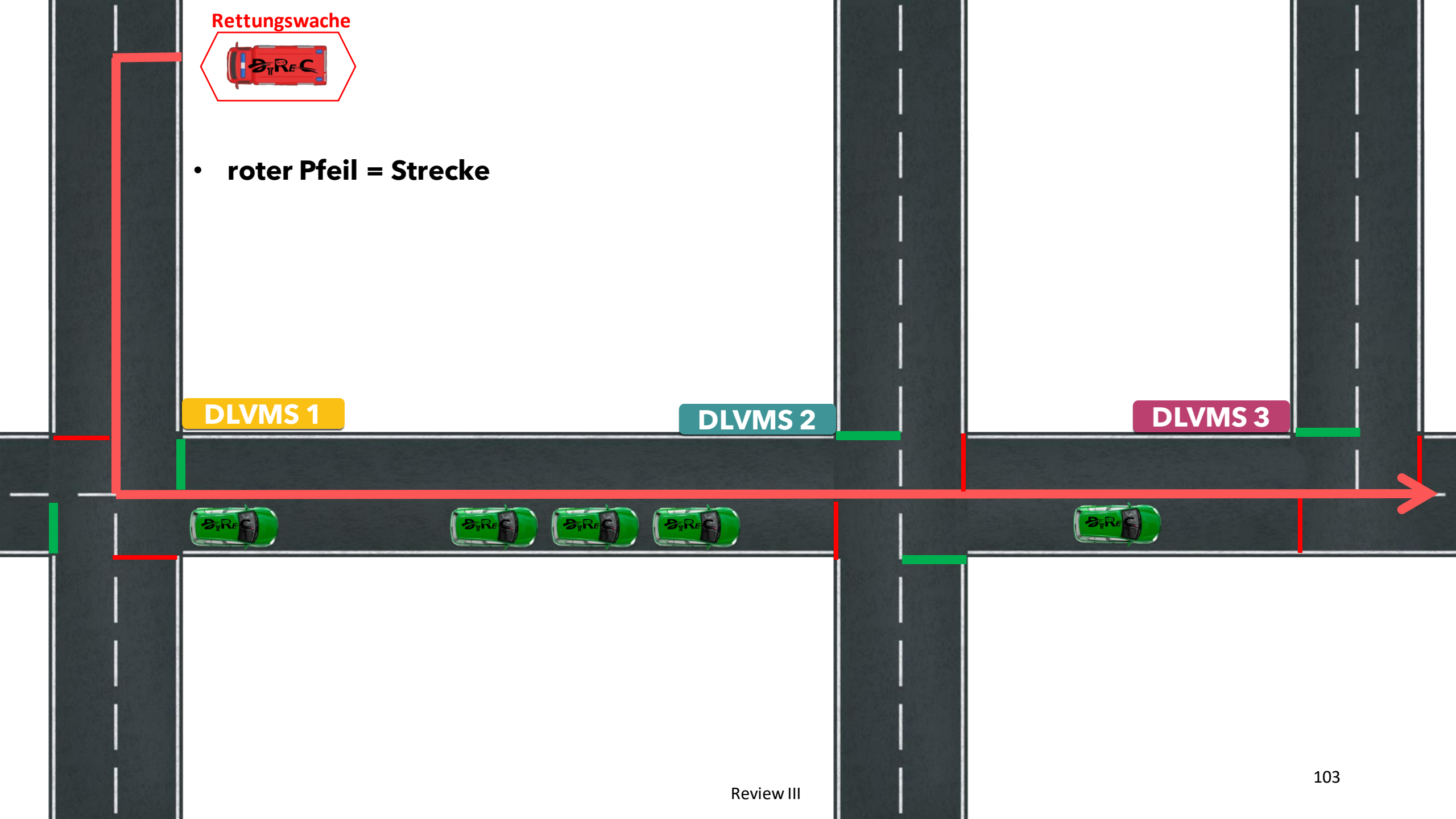


Rettungsgasse mit DyReC

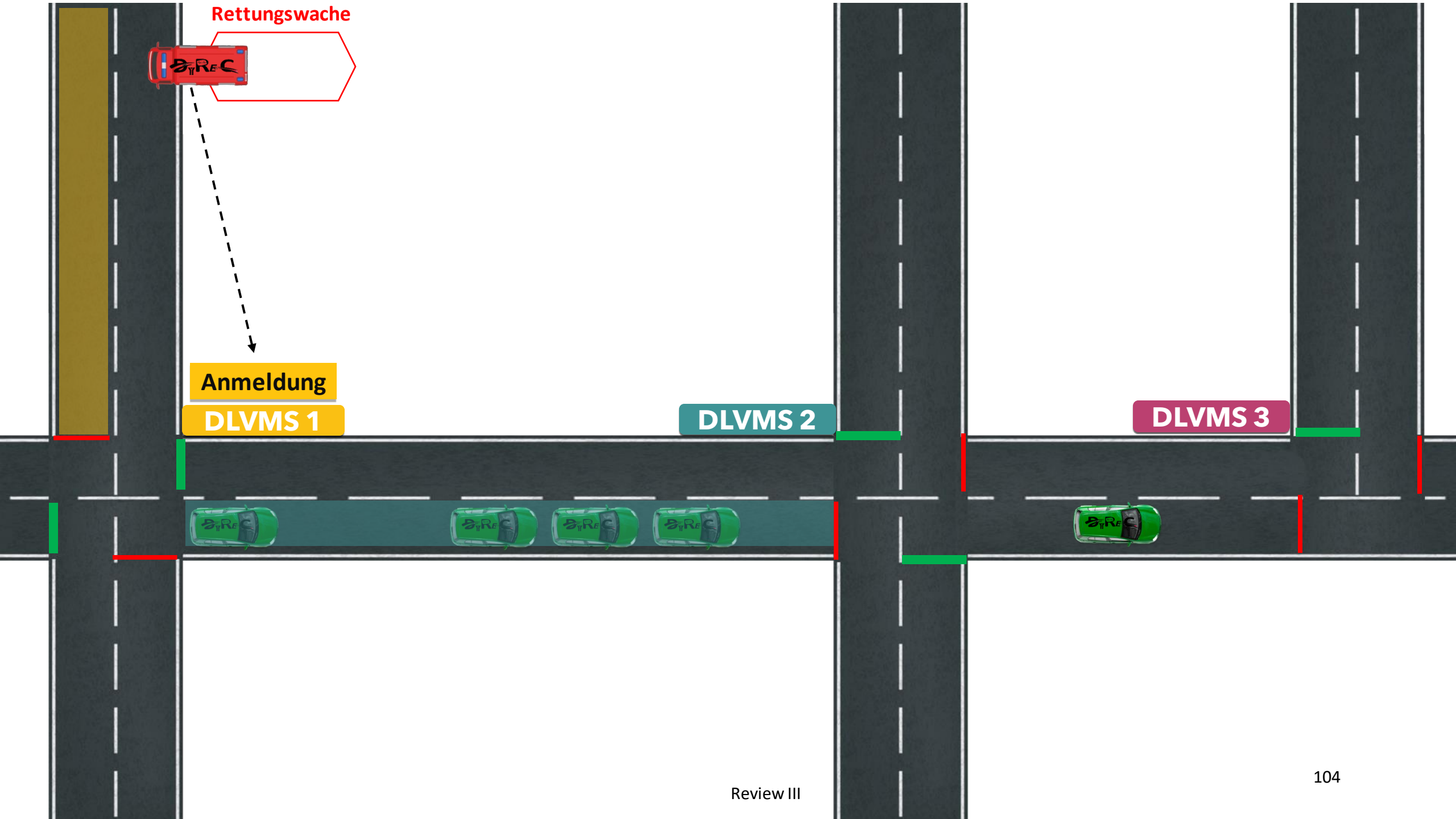
Konfliktklassifikation der Fahrzeuge

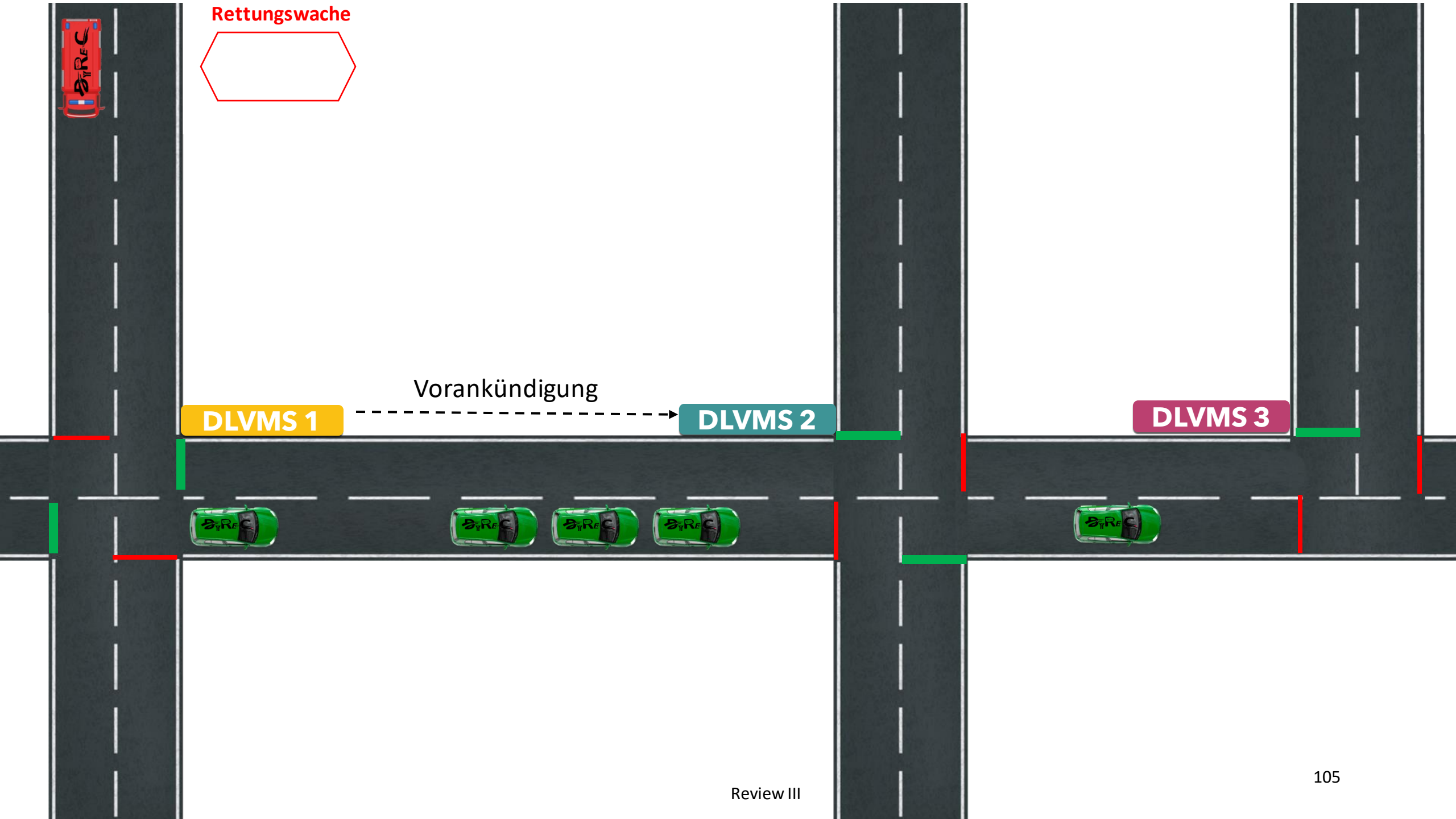


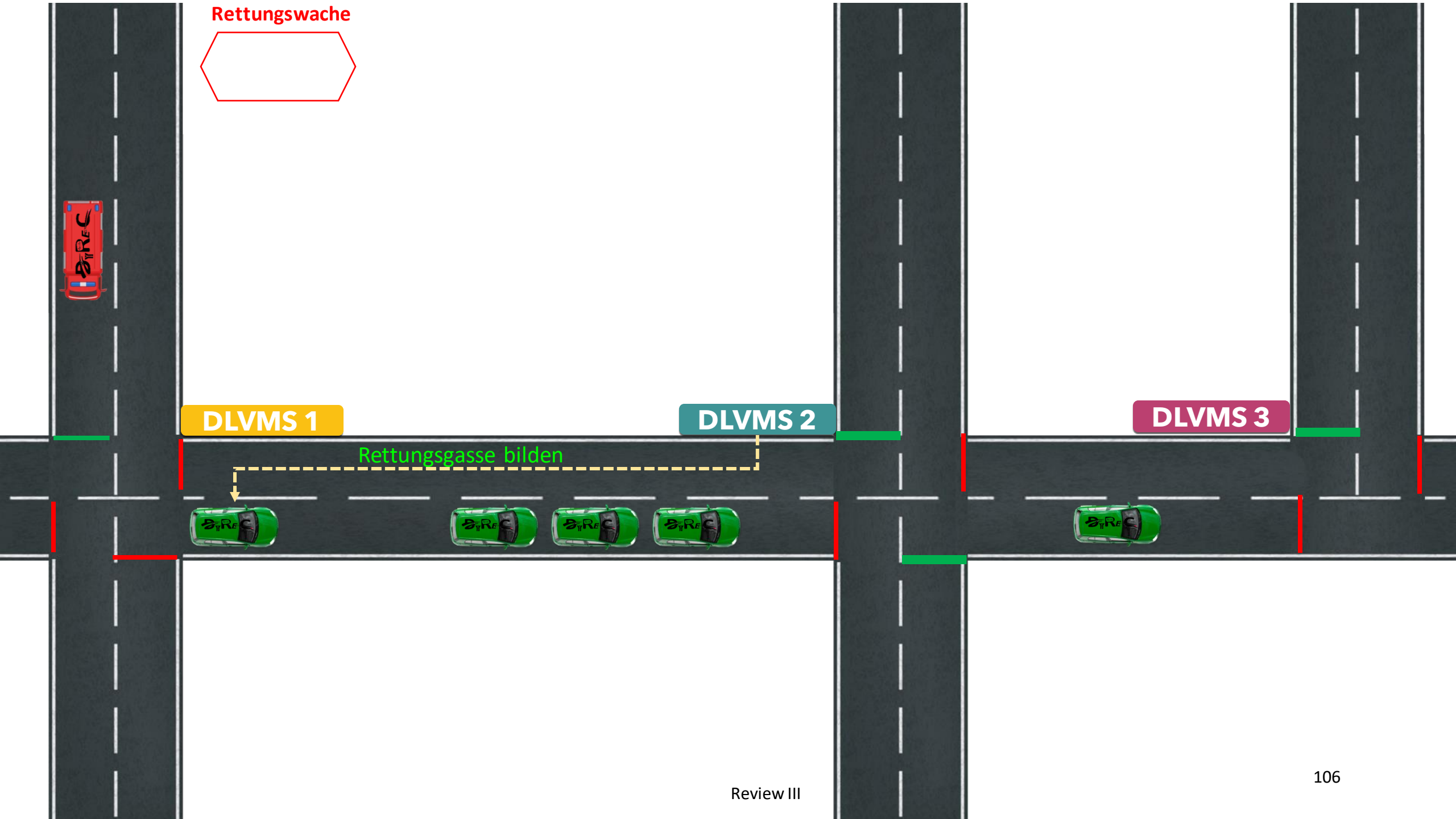
0 - Direkter Konflikt
1 - Querverkehr
2 - Abbiegender Querverkehr
3 - Verlassender direkter Verkehr
4 - zukünftiger/potentieller Konflikt
5 - kein Konflikt

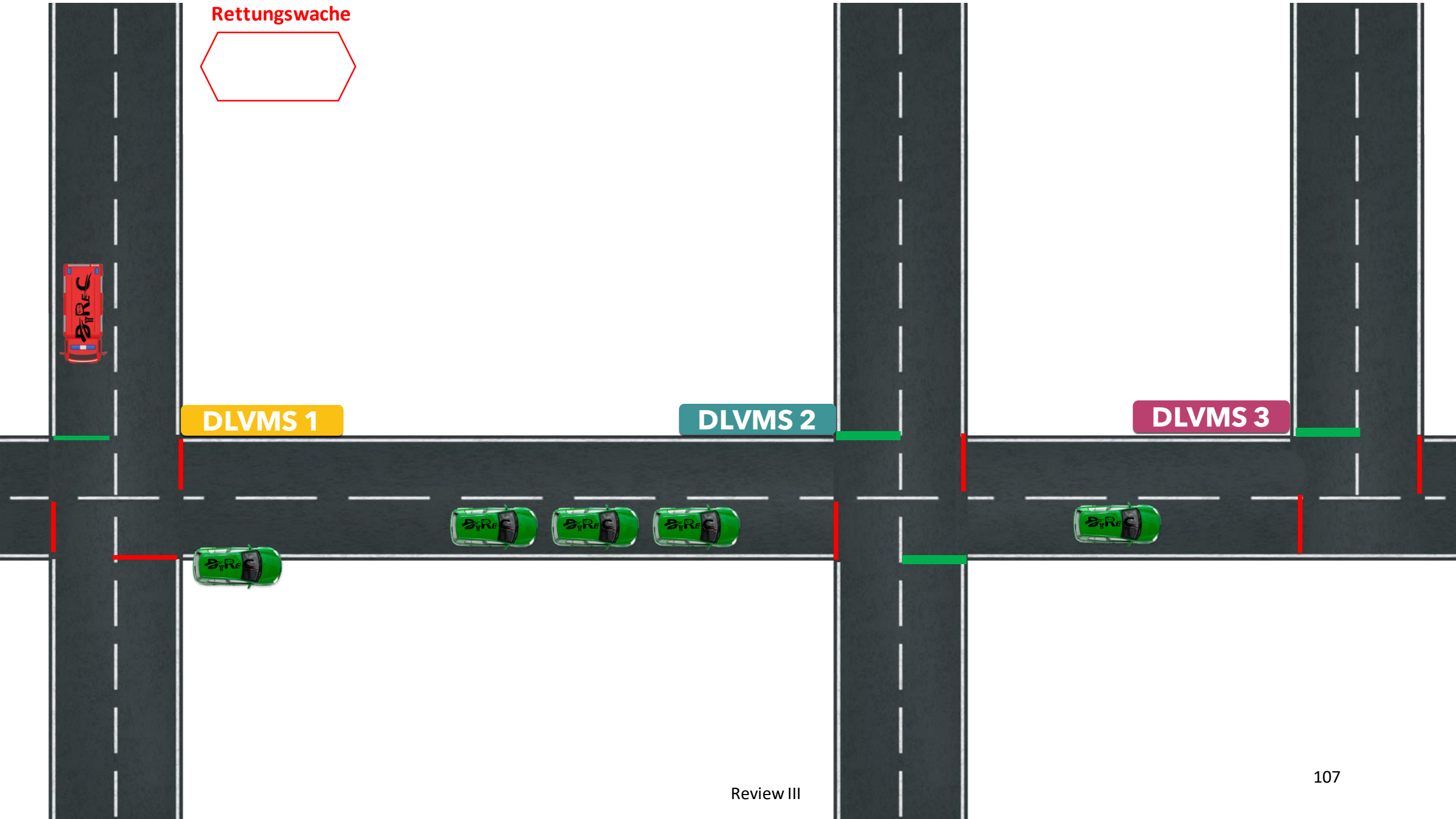


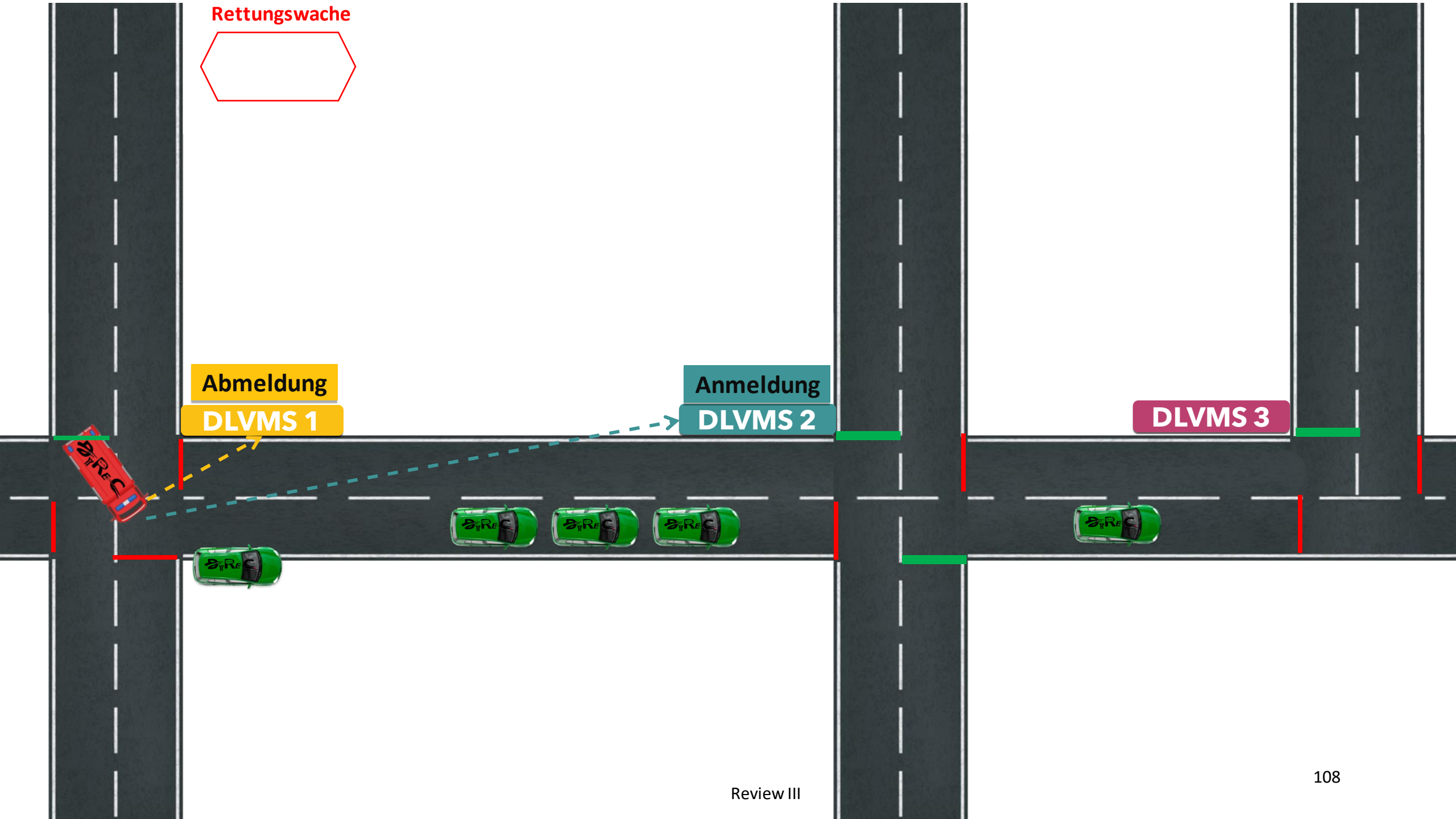
- **roter Pfeil = Strecke**

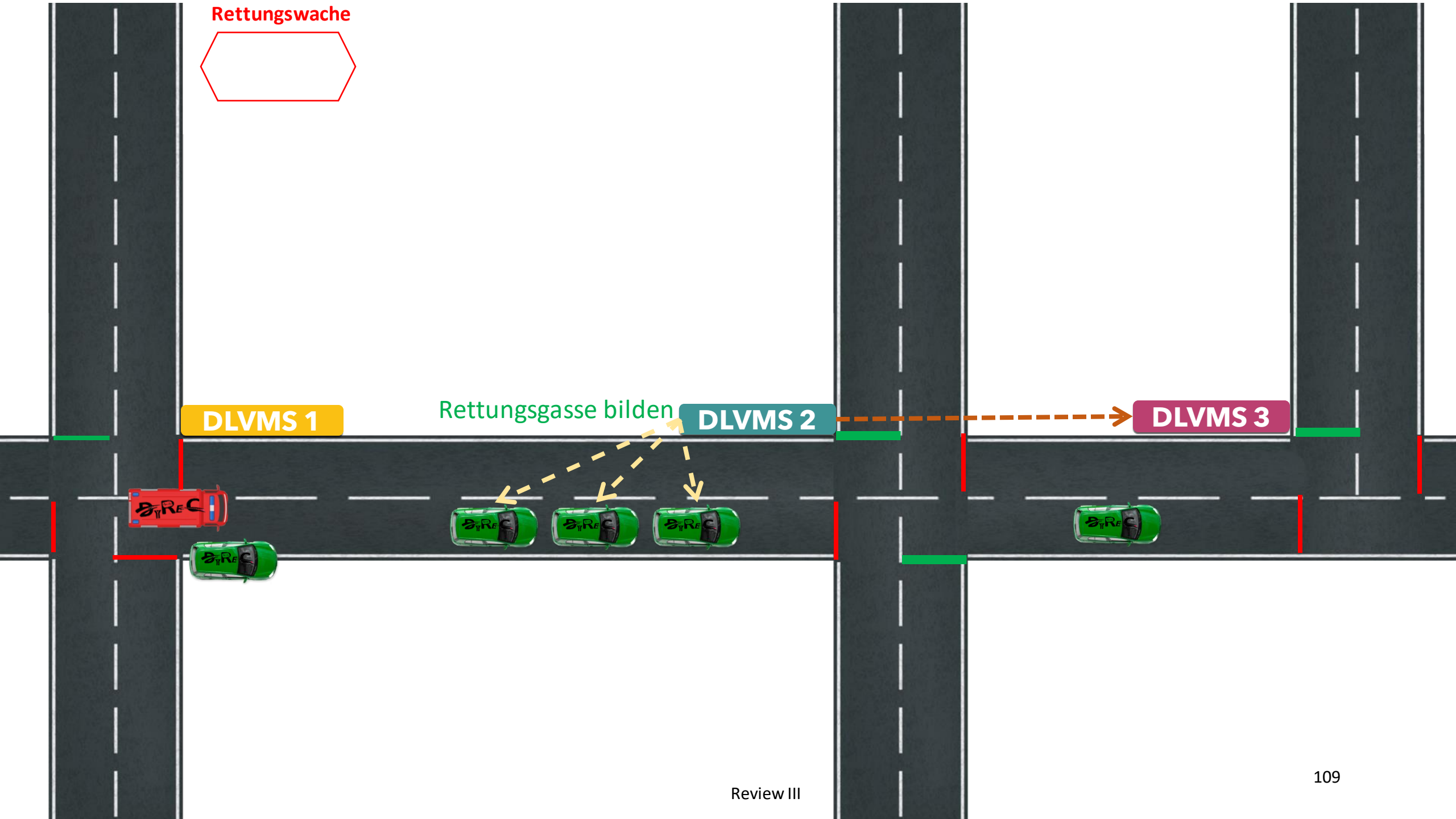

Rettungswache
Anmeldung
DLVMS 1
DLVMS 2
DLVMS 3

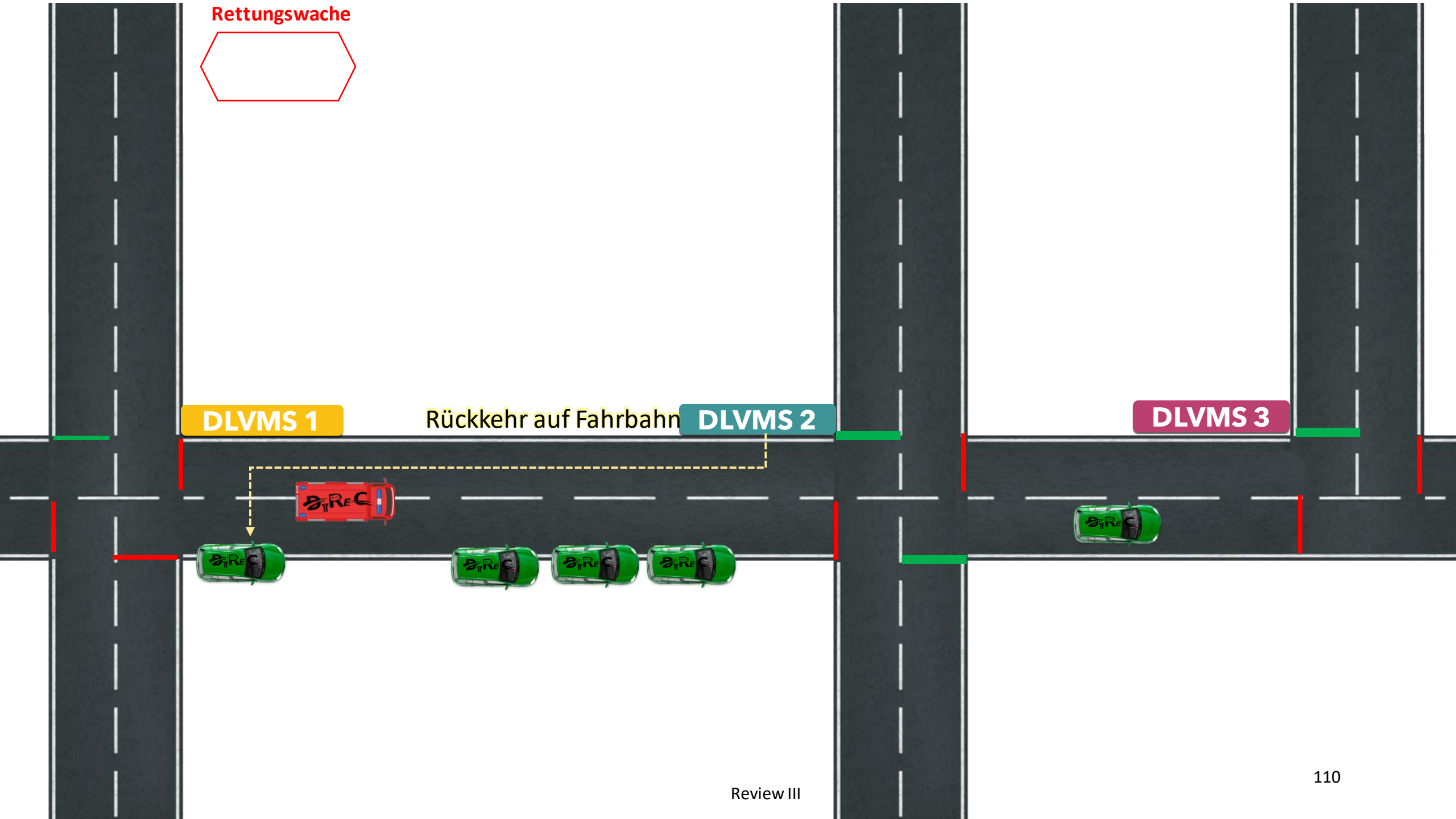

Rettungswache
Vorankündigung
DLVMS 1
DLVMS 2
DLVMS 3

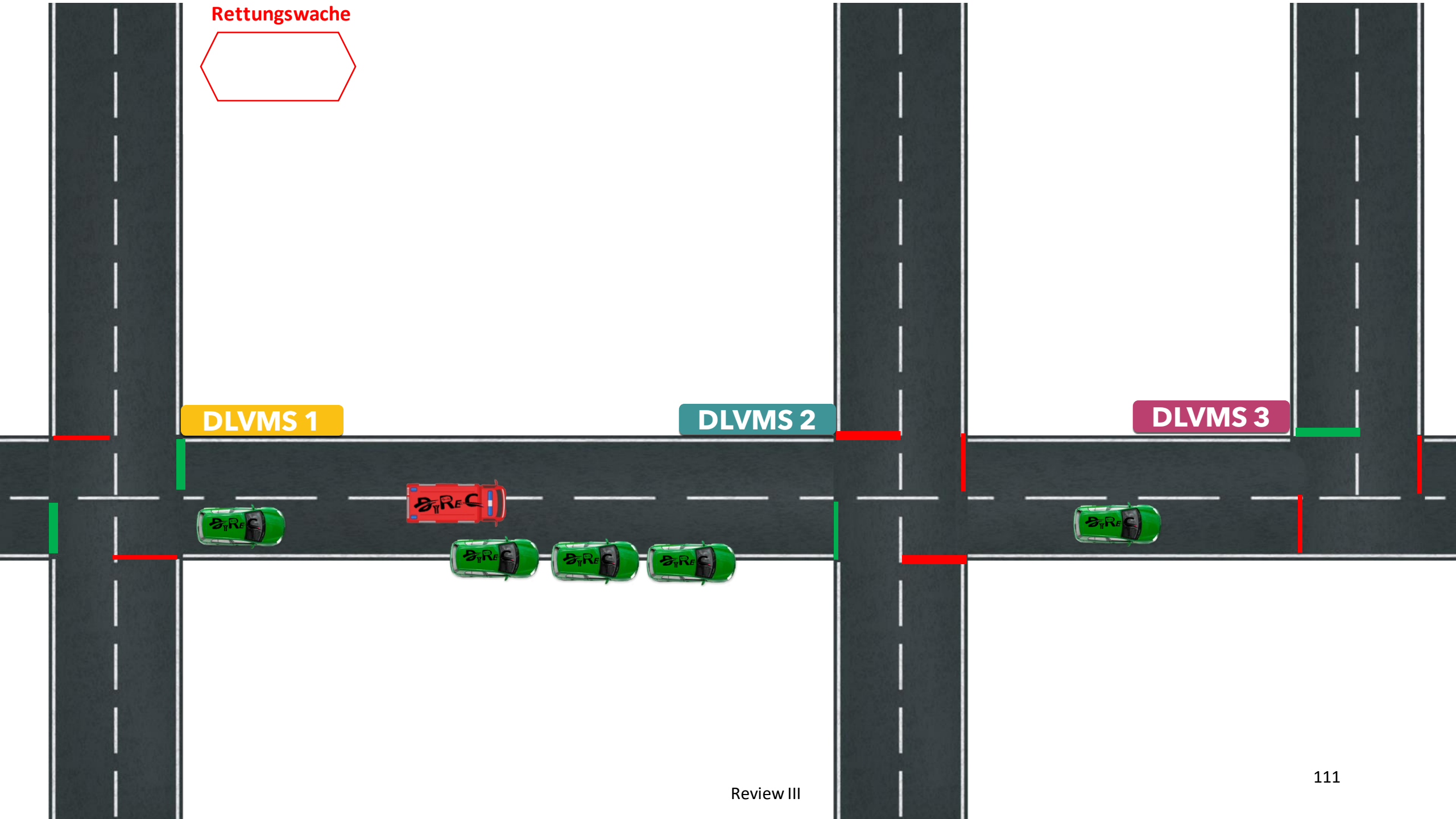

Rettungswache
DLVMS 1
DLVMS 2
DLVMS 3
Rettungsgasse bilden

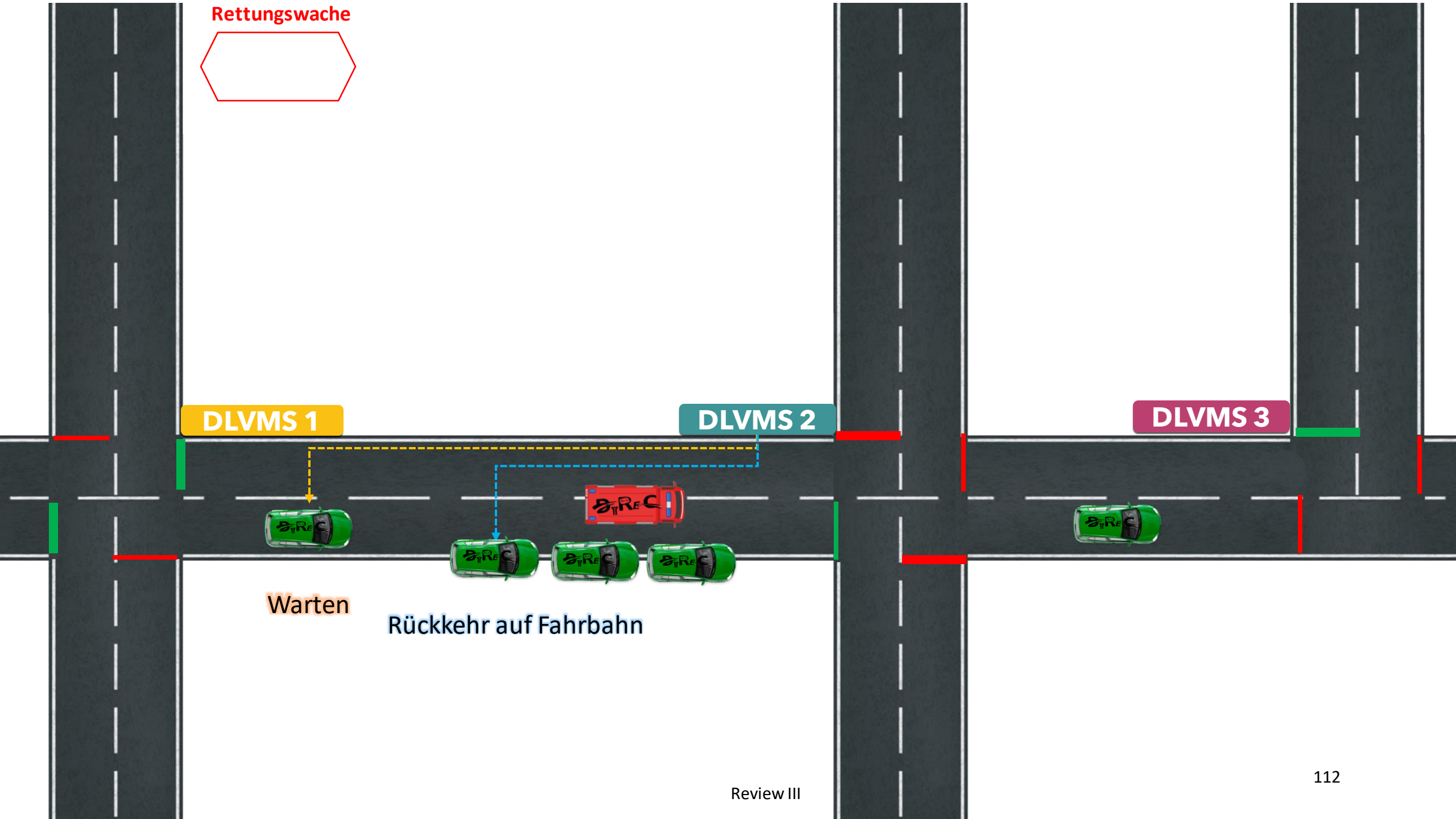

Rettungswache
DLVMS 1
DLVMS 2
DLVMS 3

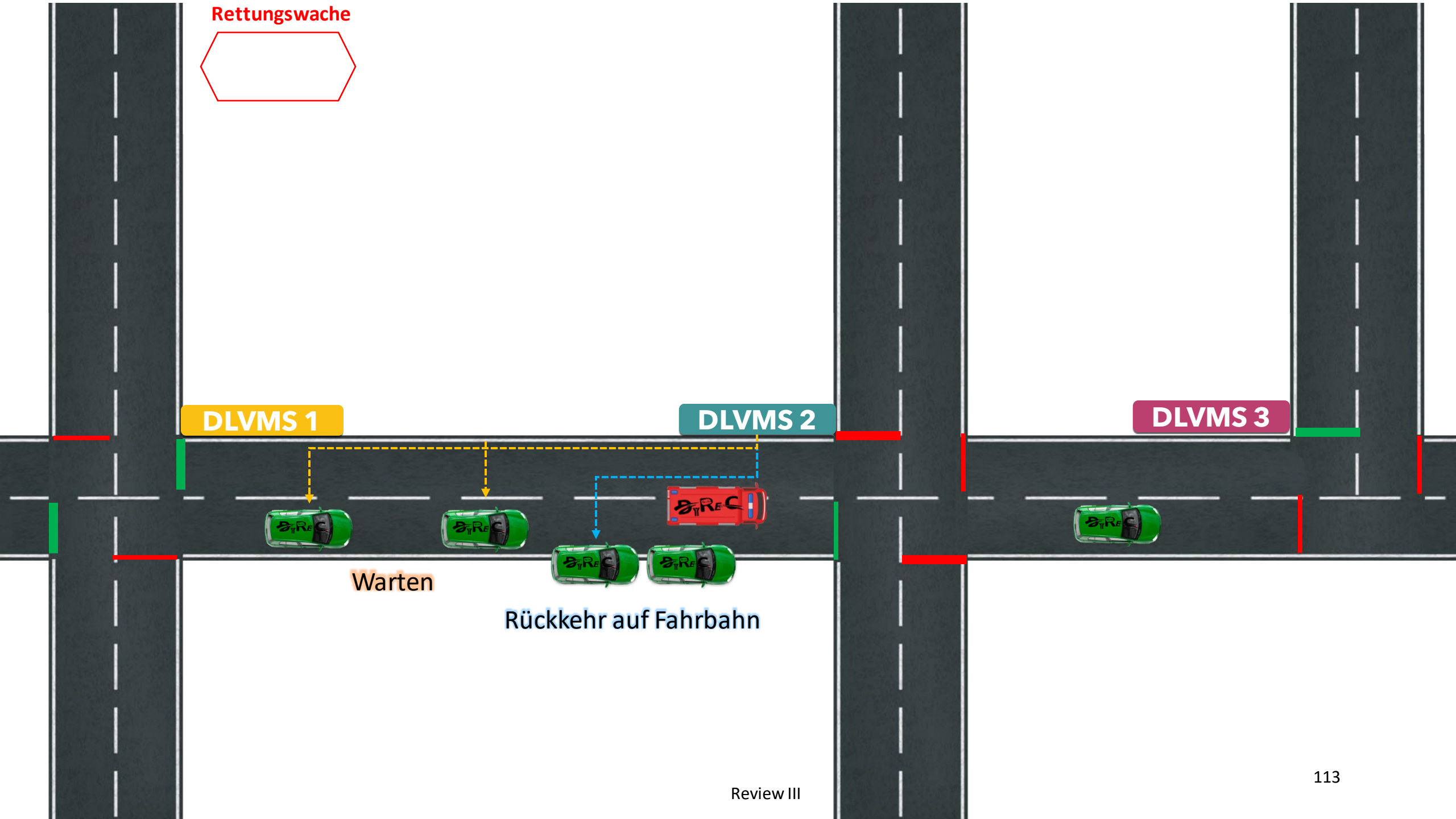

Rettungswache
Abmeldung
DLVMS 1
Anmeldung
DLVMS 2
DLVMS 3

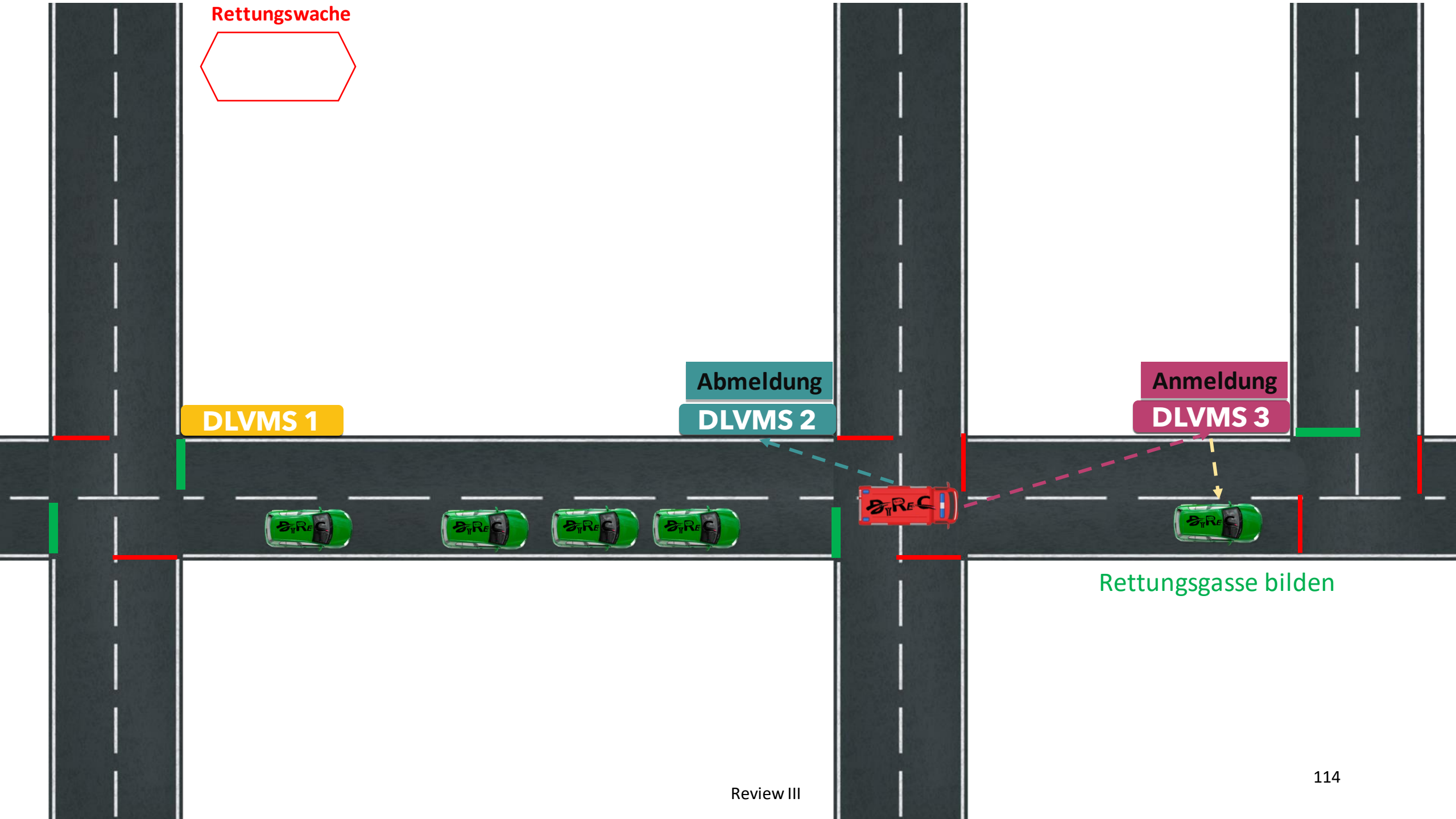

Rettungswache
DLVMS 1
Rettungsgasse bilden
DLVMS 2
DLVMS 3

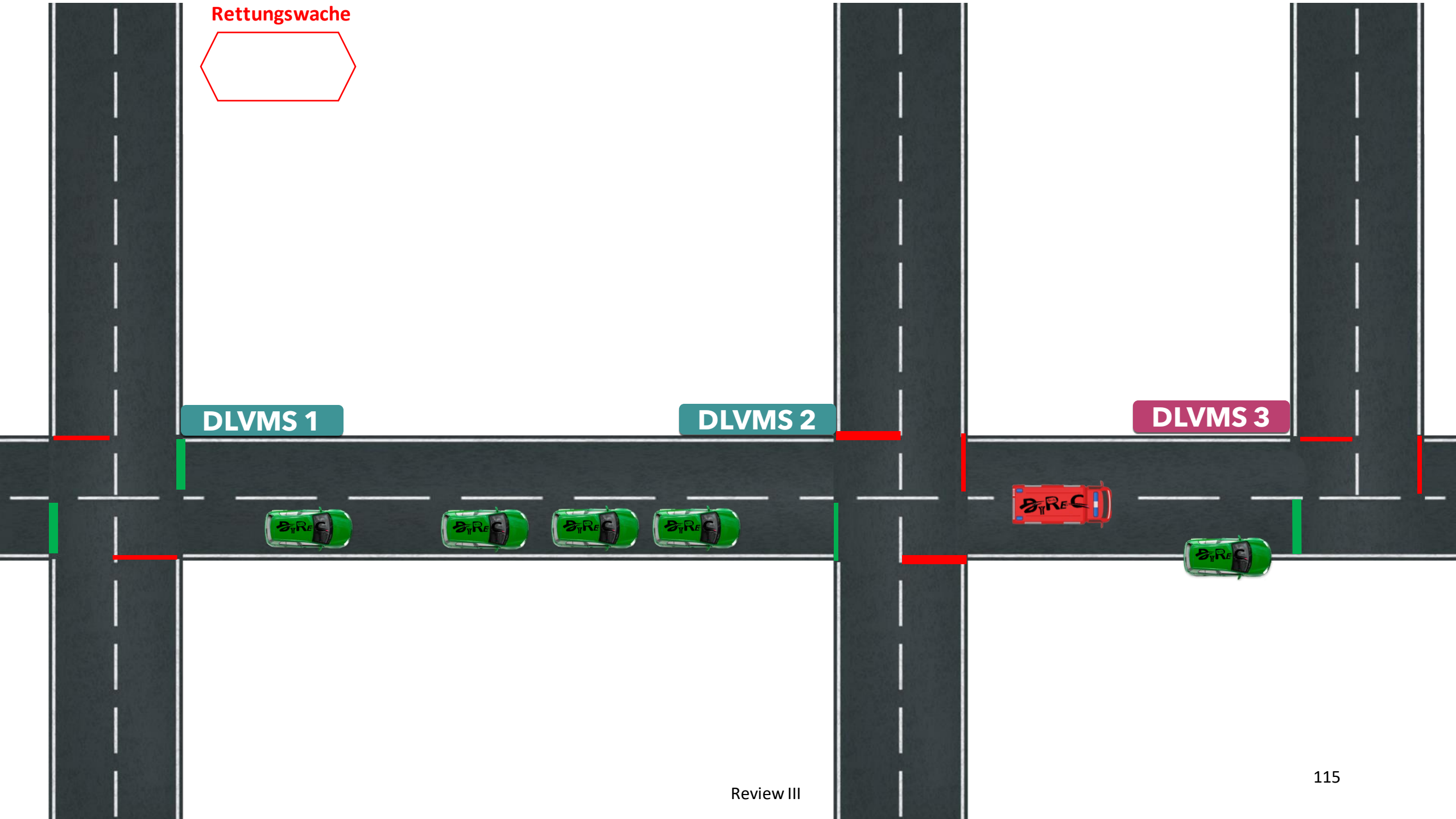

Rettungswache
DLVMS 1
Rückkehr auf Fahrbahn
DLVMS 2
DLVMS 3


Rettungswache
DLVMS 1
DLVMS 2
DLVMS 3


Rettungswache
DLVMS 1
DLVMS 2
DLVMS 3
Warten
Rückkehr auf Fahrbahn


Rettungswache
DLVMS 1
DLVMS 2
DLVMS 3
Warten
Rückkehr auf Fahrbahn


Rettungswache
DLVMS 1
Abmeldung
DLVMS 2
Anmeldung
DLVMS 3
Rettungsgasse bilden


Rettungswache
DLVMS 1
DLVMS 2
DLVMS 3

Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
  - Referenzsystem
  - Kaskadierendes Verfahren
  - LSA
  - Dynamische Rettungsgasse
  - KI-Konzept
- Implementierung
- Test
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Motivation

Ziel: gezielte Manipulation des Verkehrsflusses

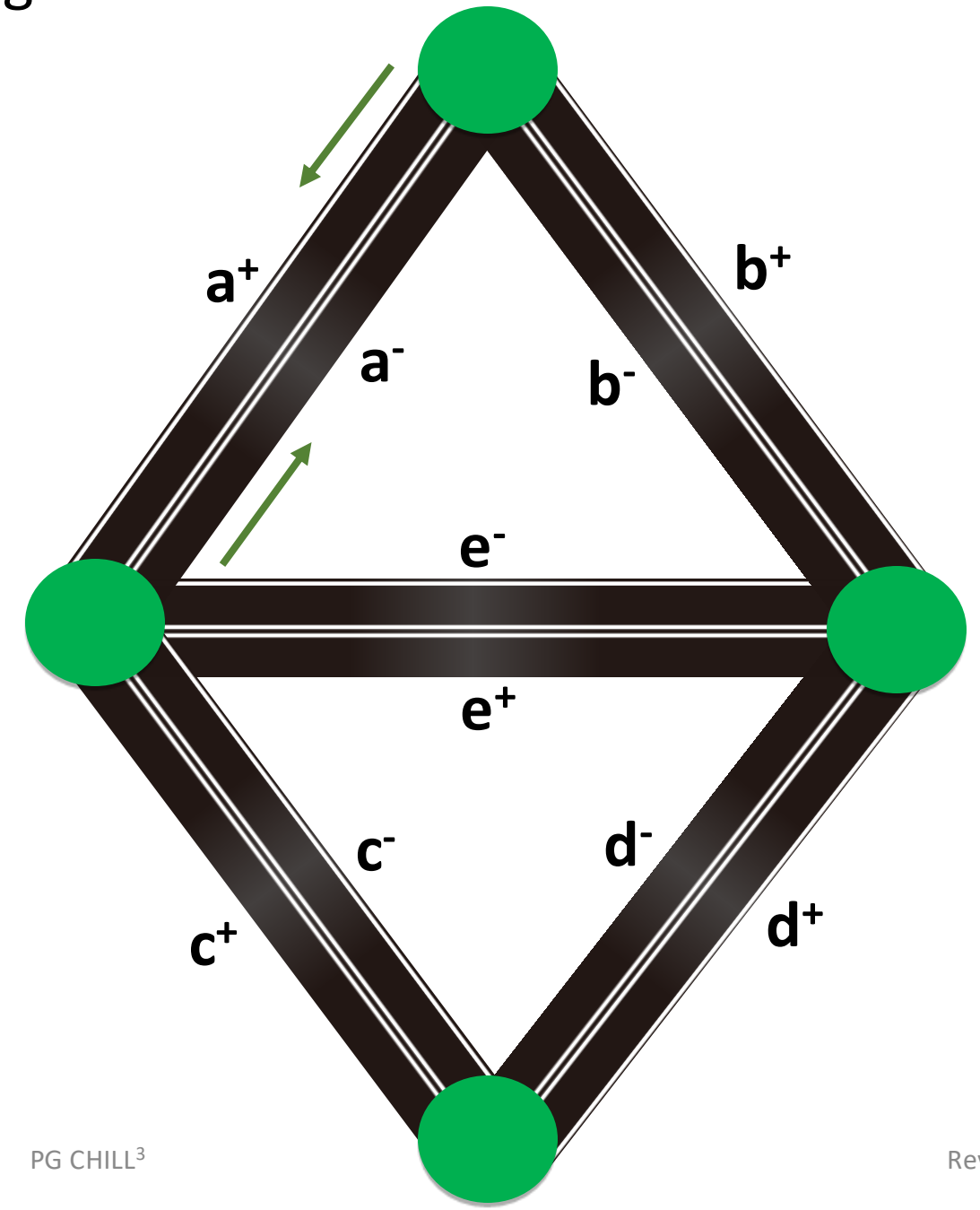
🎯 Bevorrechtigen von Fahrzeugen

🎯 Gesamtdurchsatz

Anforderung:

- **einfaches** Modell des Verkehrsflusses zur **Vorhersage** und Optimierung des Verkehrsflusses

Spurgenaues Straßennetz



	a^+	a^-	b^+	b^-
a^+	1	0	1	0
a^-	0	1	0	0
b^+	0	0	1	0
b^-	0	1	0	1

Adjazenzmatrix A

Von Adjazenz zu Wahrscheinlichkeiten

Ersetze in $A=(a_{ij})$ die Einträge durch Wahrscheinlichkeiten:

- a_{ij} ist die Wahrscheinlichkeit für ein Fahrzeug in einem Zeitschritt von der Spur i zur Spur j zu wechseln
- Diagonalelement a_{ii} ist die Wahrscheinlichichkeit für ein Fahrzeug auf der Spur i zu bleiben

Die Einträge der Matrix A(t) sollen mittels eines gleitenden Durchschnitts erlernt werden.

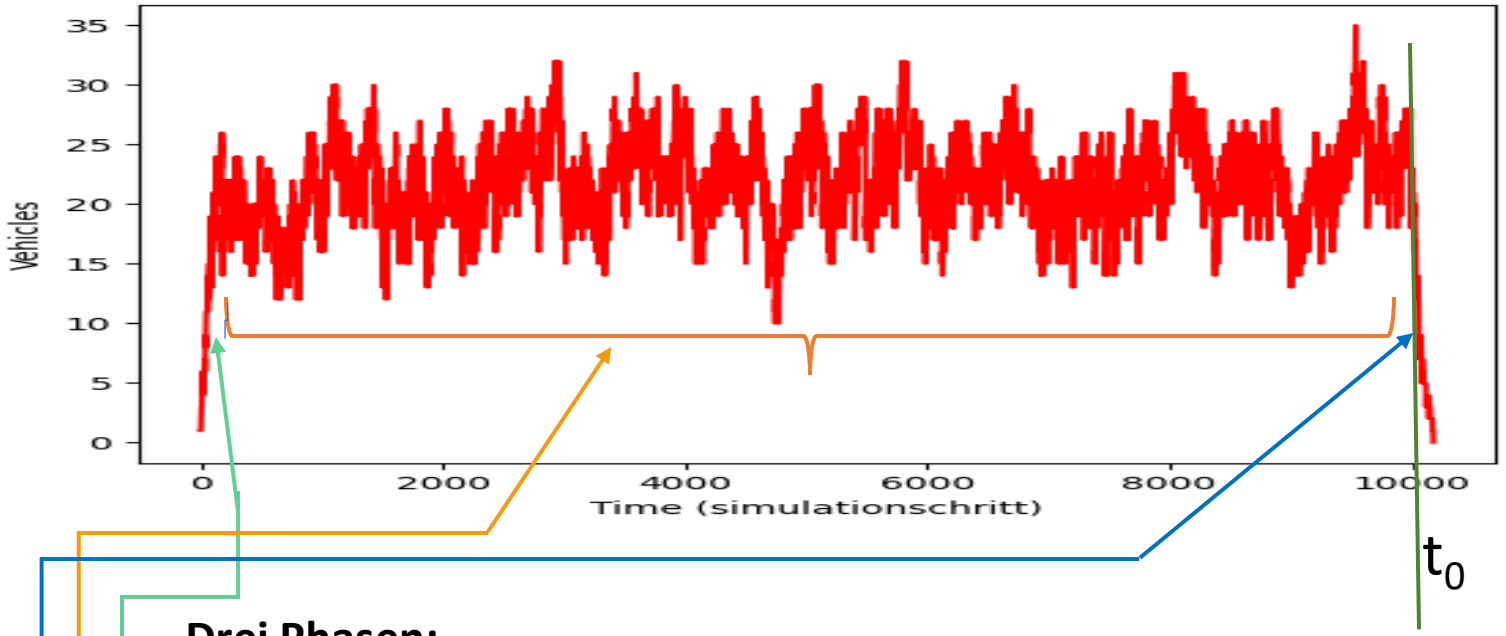
Der Vektor v(t) gibt die Verteilung der Fahrzeuge zum Zeitschritt t an.
(Σv_i = Anzahl der Fahrzeuge zum Zeitpunkt t.)

Basismodell
$v(t+1) = A(t)*v(t)$

Proof of Concept

Kann das Modell den Verkehrsfluss angemessen darstellen?

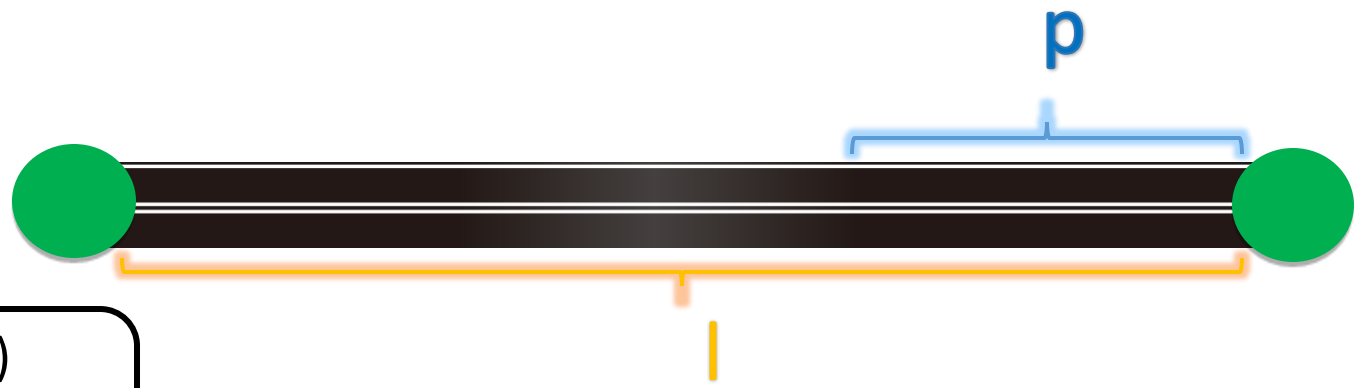
Vergleich mit der mikroskopischen Simulation in SUMO



Drei Phasen:

1) **Aufbauphase:** es werden zeitlich gleichverteilt Fahrzeuge generiert
2) **Haltephase** : Generation und Verschwinden der Fahrzeuge halten sich die Waage
3) **Abbauphase** : Ab t_0 werden keine Fahrzeuge mehr generiert

Adaption des Verkehrflussmodels



A ist zeitlich konstant, kann aus Turn Ratios (SUMO) mithilfe des zusätzlichen Modellparameters p berechnet werden

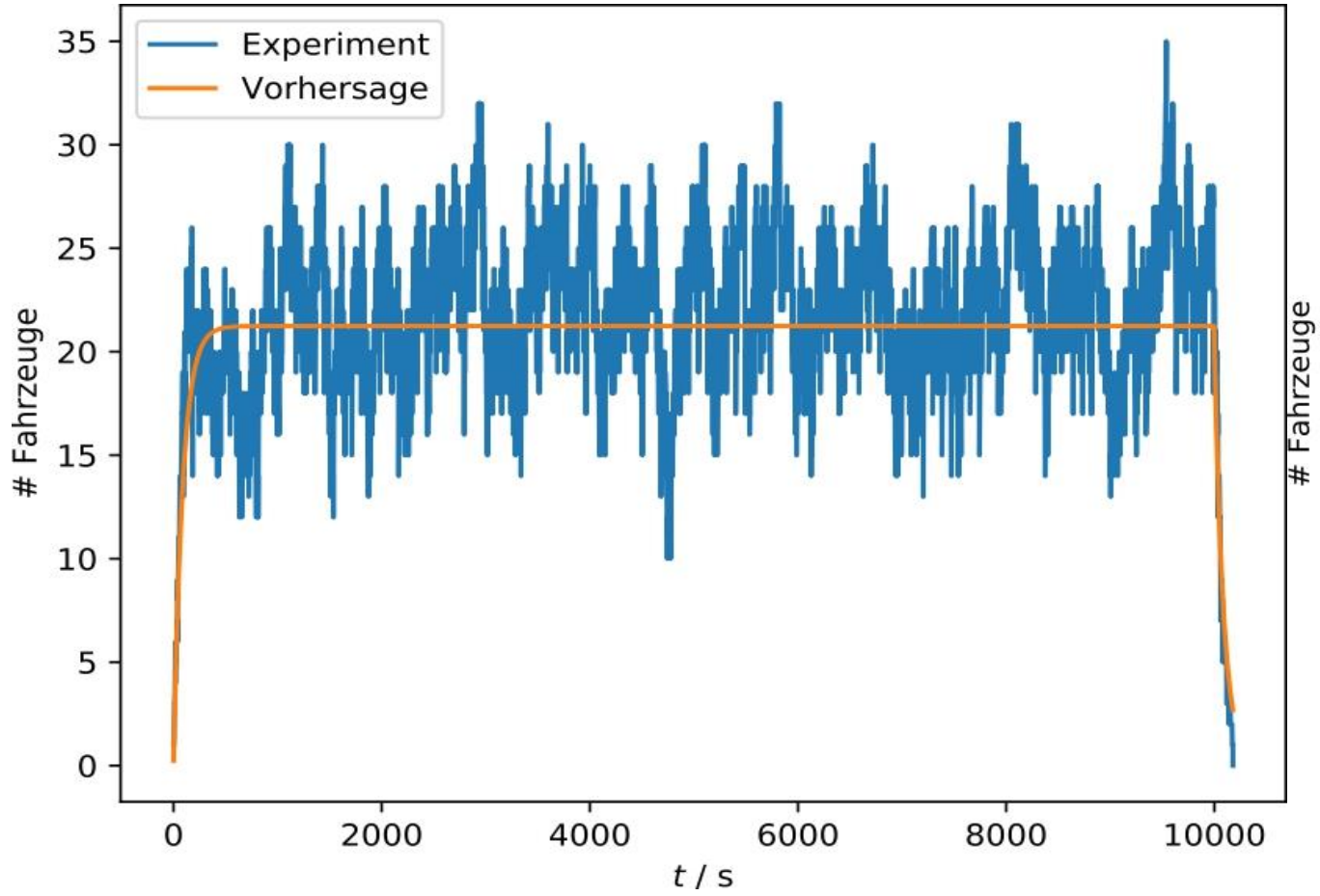
p/l ist die Wahrscheinlichkeit dafür, dass ein Fahrzeug die Spur verlässt

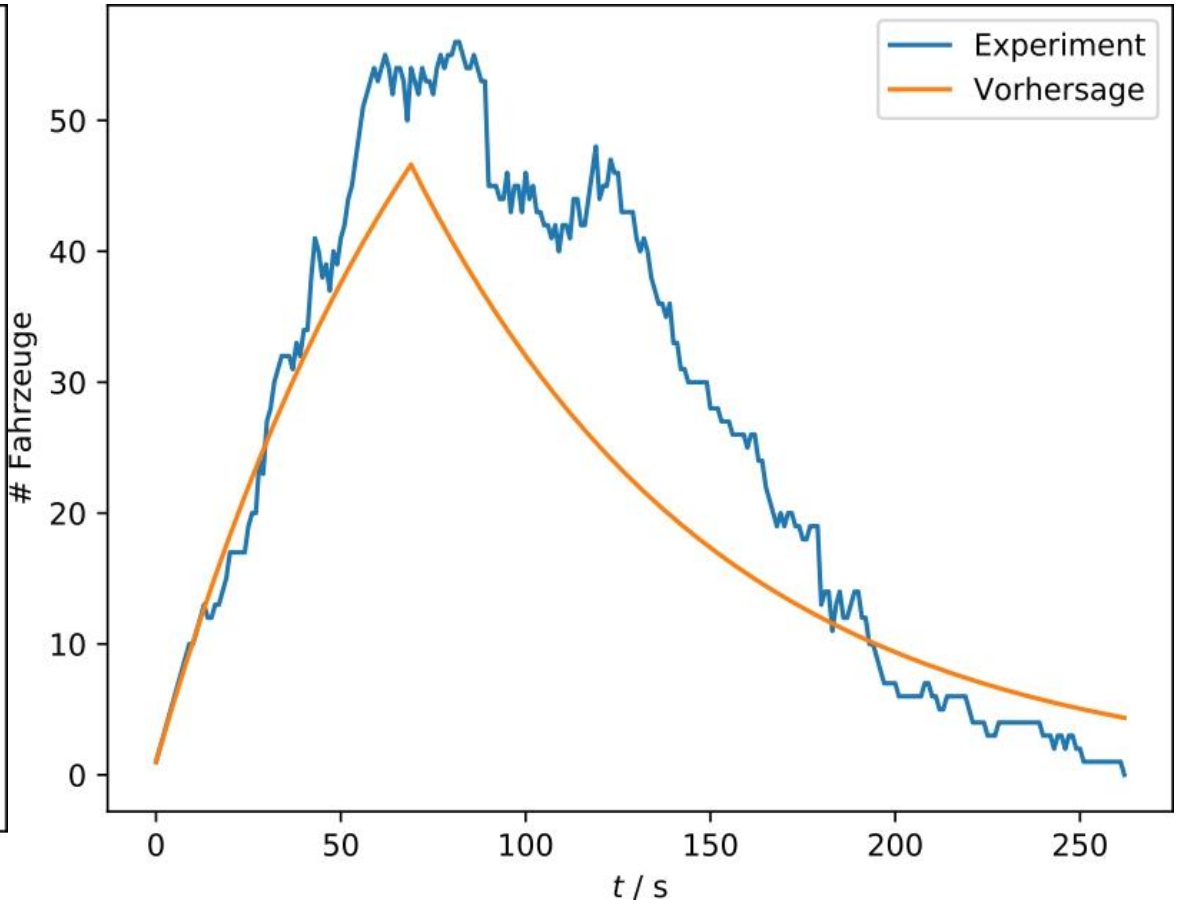
Der konstante Vektor b sorgt für die gleichverteilte Erzeugung von Fahrzeugen, b lässt sich aus der Routen-Datei (SUMO) berechnen

$$\vec{v}(t+1) = A \cdot \left[\vec{v}(t) + \vec{b}\right] \quad \text{falls } t <= t0$$

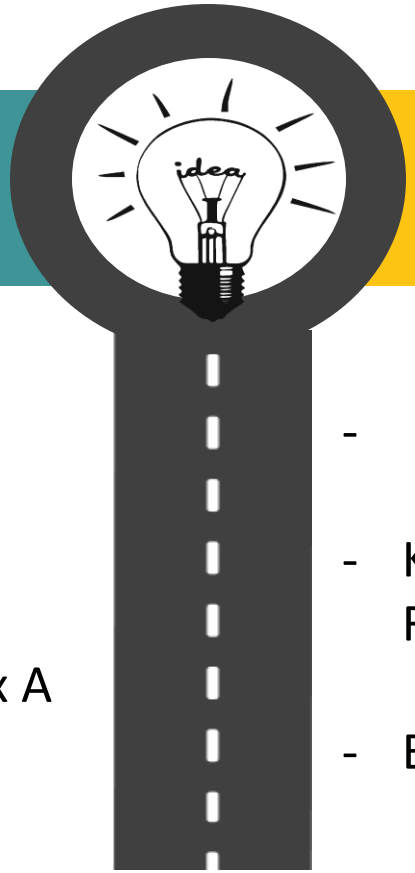
$$\vec{v}(t+1) = A \cdot \vec{v}(t) \quad \text{falls } t0 < t$$

KI-Konzept:


Experiment
Vorhersage
Fahrzeuge
t / s
0
5
10
15
20
25
30
35
0
2000
4000
6000
8000
10000


Experiment
Vorhersage
Fahrzeuge
t / s
0
10
20
30
40
50
0
50
100
150
200
250

Fazit :



- Indizien für ausreichende Modellgüte
- p=6 scheint eine gute Wahl
- Ampelsteuerung kompatibel mit Matrix A

Weitere Schritte :

- Erlernen der Matrix A
- Kombination mit Ampelsteuerung und Formulierung eines Optimierungsproblems
- Einbindung in DyReC

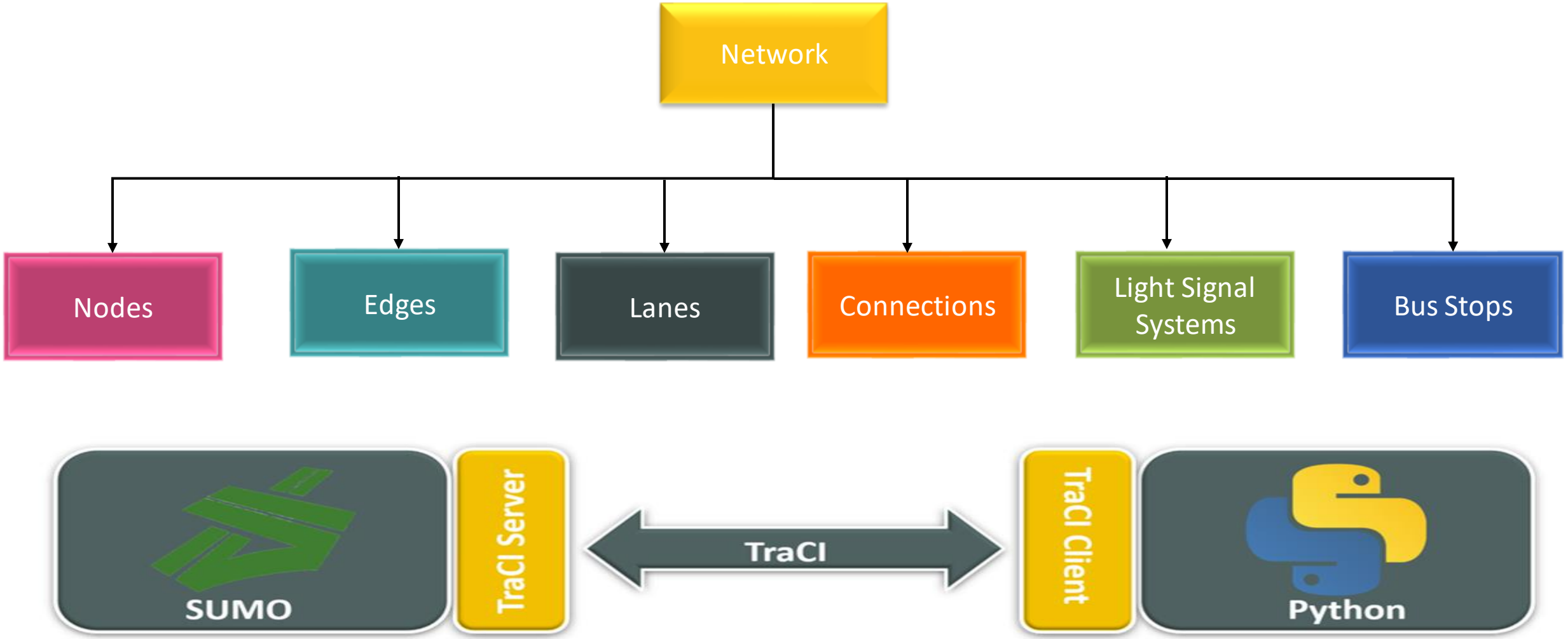
Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
  - SUMO-Umgebung
  - Komponentendiagramm
  - Programmablauf
  - Kaskadierendes Verfahren
  - LSA-Anpassung
  - Dynamische Rettungsgasse
  - Tests
- Demonstrator
- Evaluation
- Ausblick
- Fazit

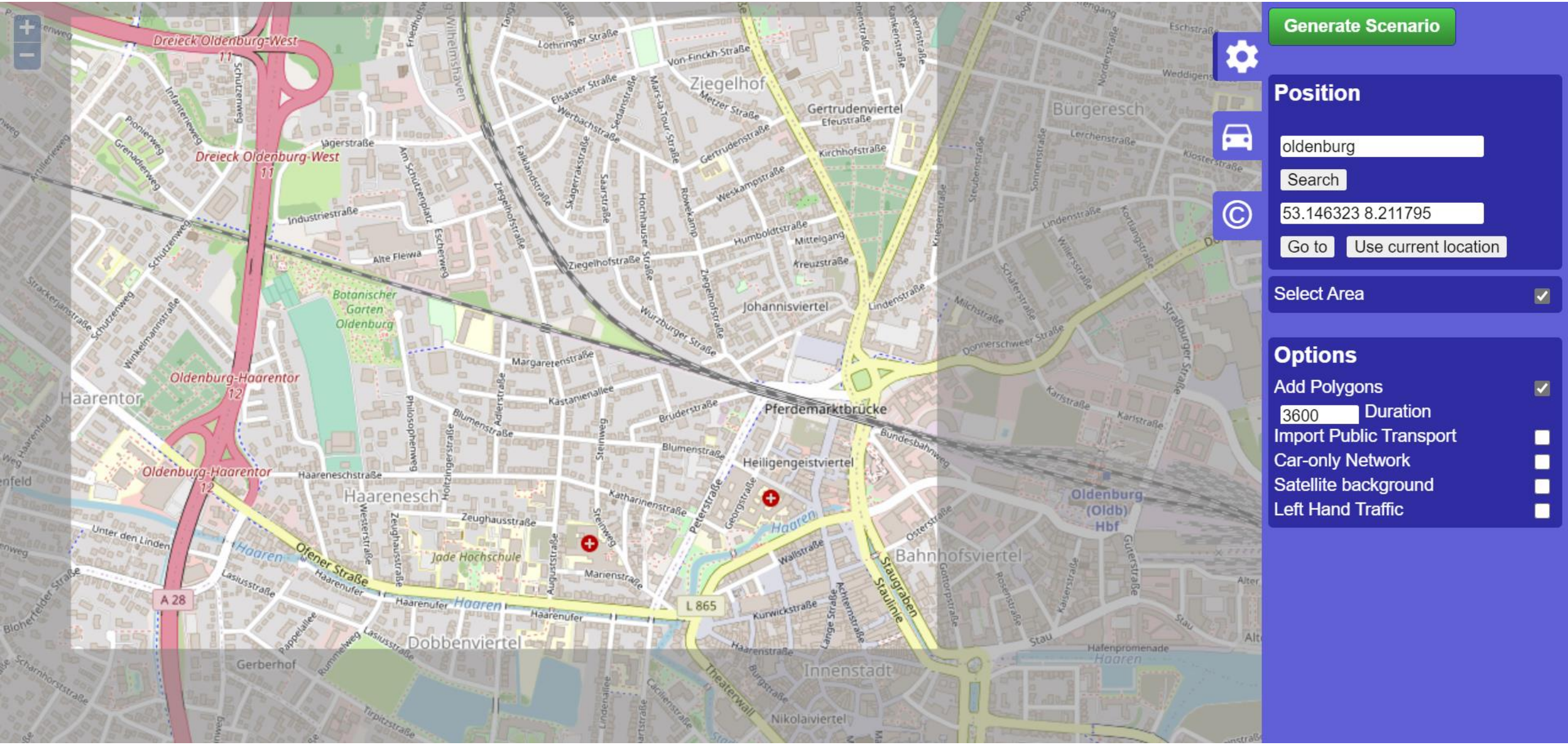
Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
  - SUMO-Umgebung
  - Komponentendiagramm
  - Programmablauf
  - Kaskadierendes Verfahren
  - LSA-Anpassung
  - Dynamische Rettungsgasse
  - Tests
- Demonstrator
- Evaluation
- Ausblick
- Fazit

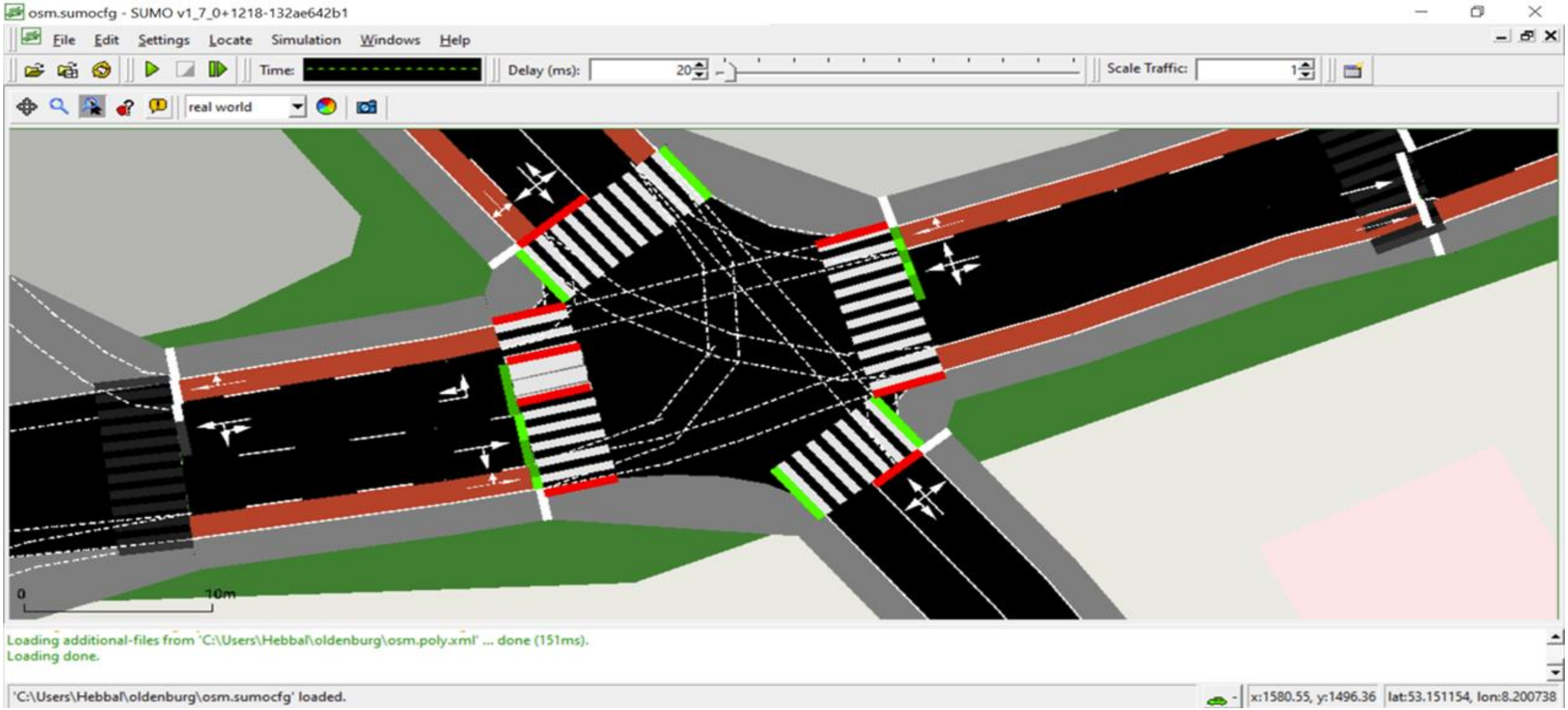
SUMO: ist eine Open-Source mikroskopische Verkehrssimulation.



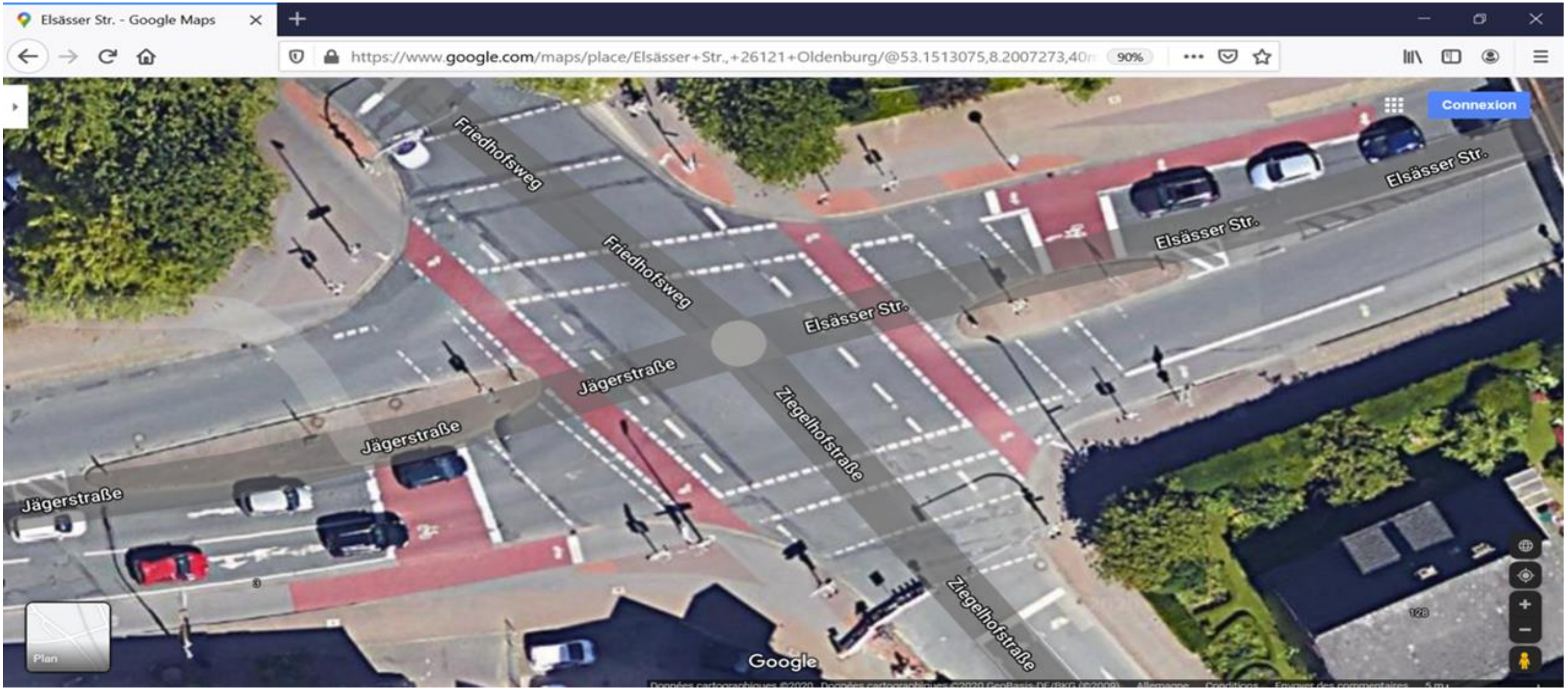
OSM Web Wizard



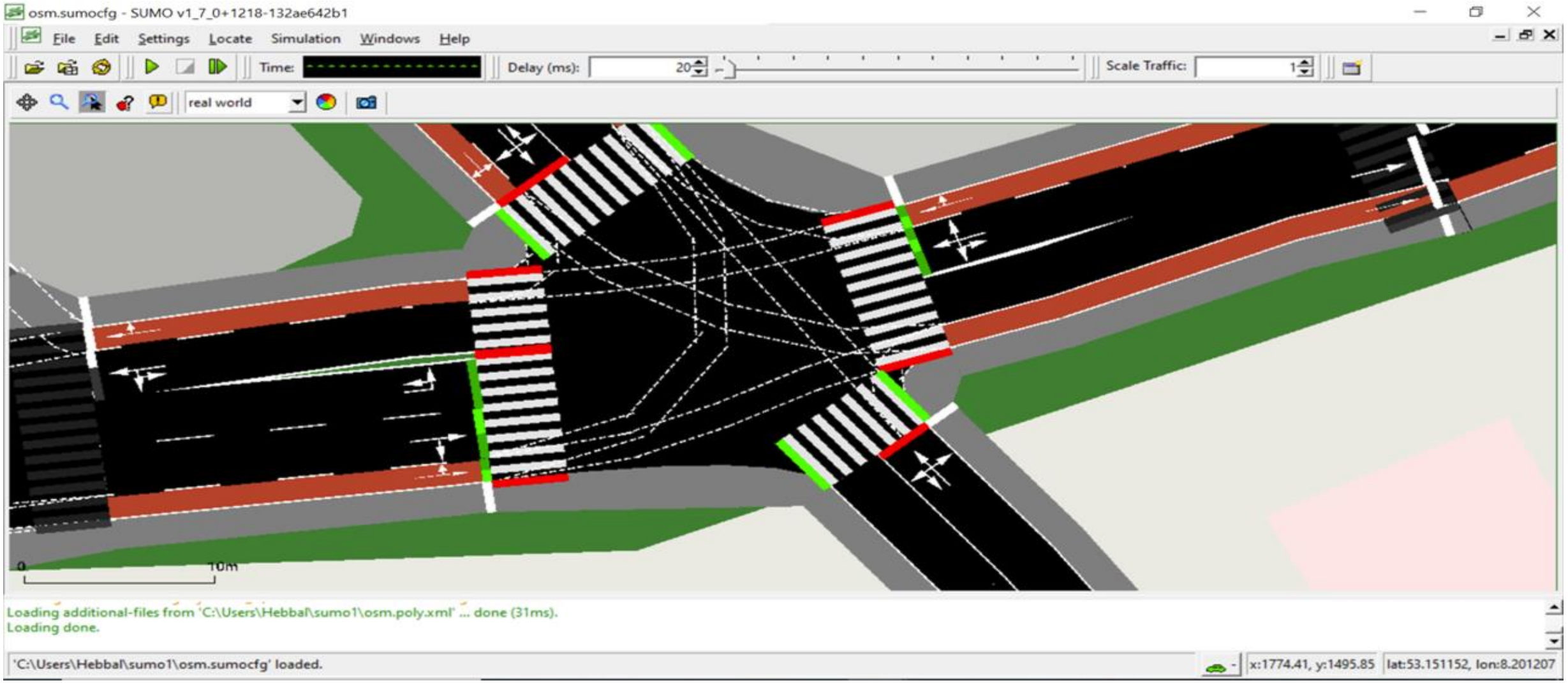
Schwierigkeiten



Schwierigkeiten



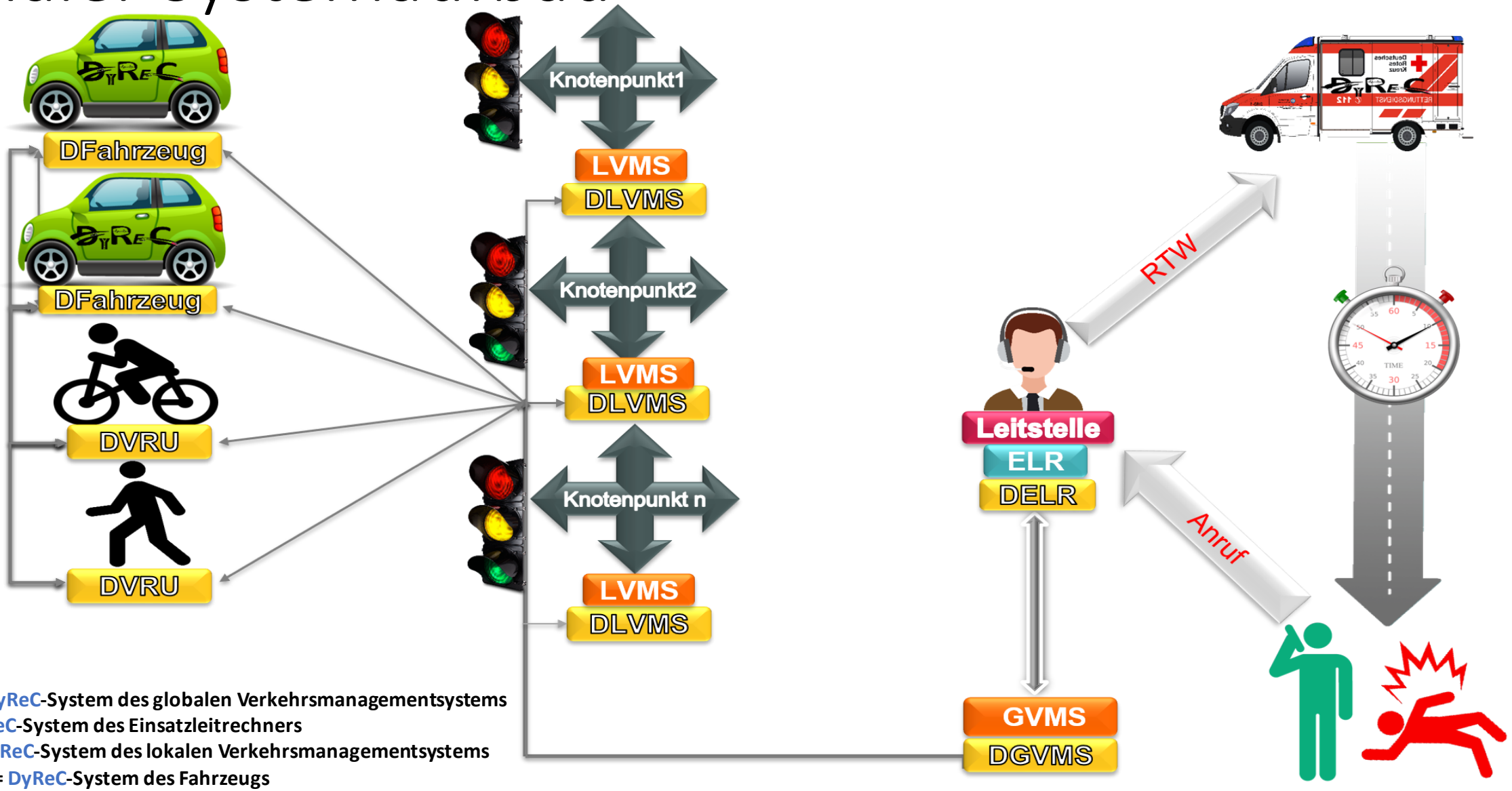
Schwierigkeiten



Agenda

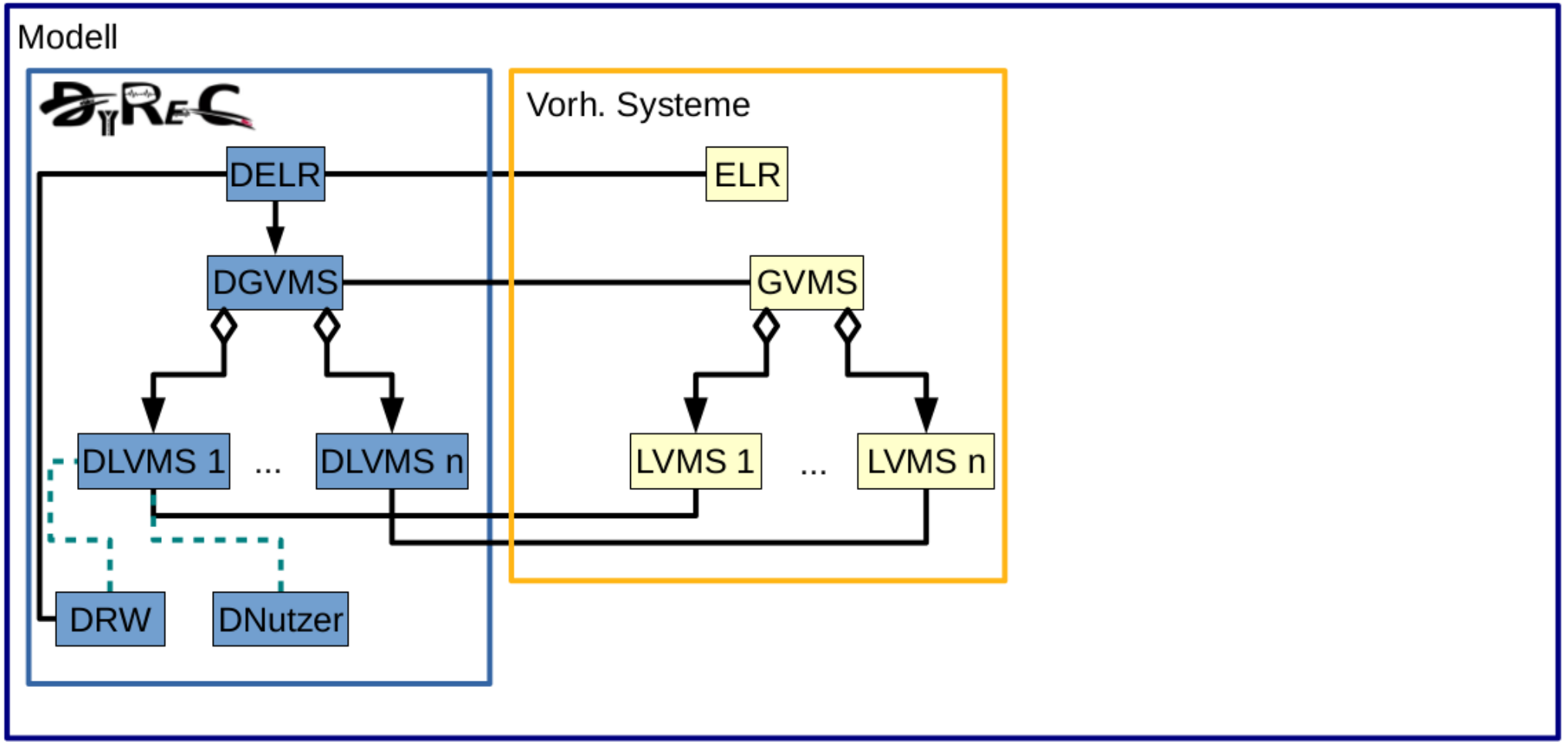
- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
  - SUMO-Umgebung
  - Komponentendiagramm
  - Programmablauf
  - Kaskadierendes Verfahren
  - LSA-Anpassung
  - Dynamische Rettungsgasse
  - Tests
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Initialer Systemaufbau

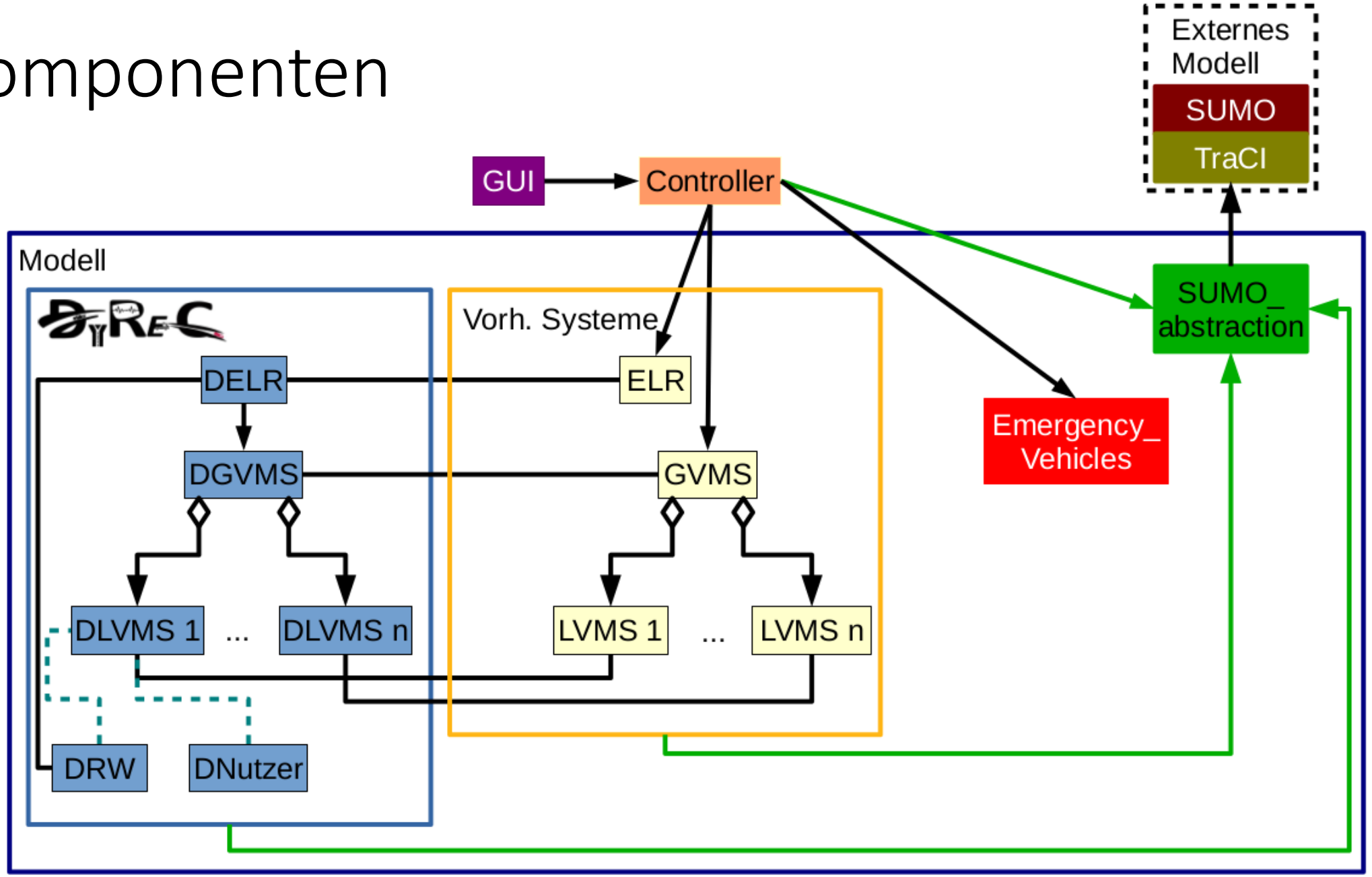


DGVMS= DyReC-System des globalen Verkehrsmanagementsystems
DELR= DyReC-System des Einsatzleitrechners
DLVMS= DyReC-System des lokalen Verkehrsmanagementsystems
DFahrzeug= DyReC-System des Fahrzeugs
DVRU= DyReC-System des VRUs (z. B. Fußgänger oder Fahrradfahrer)

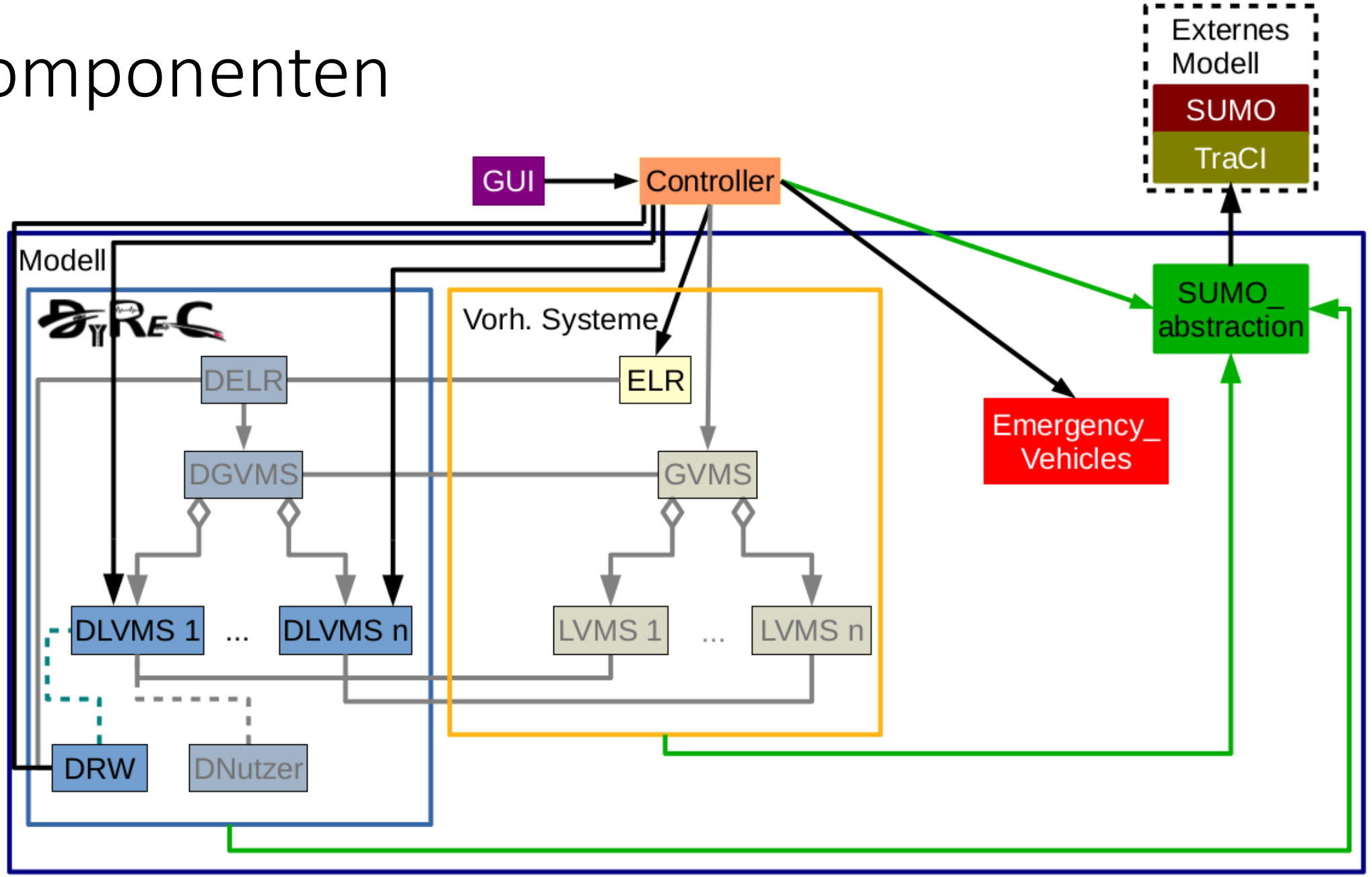
Systemkomponenten (in der realen Welt)



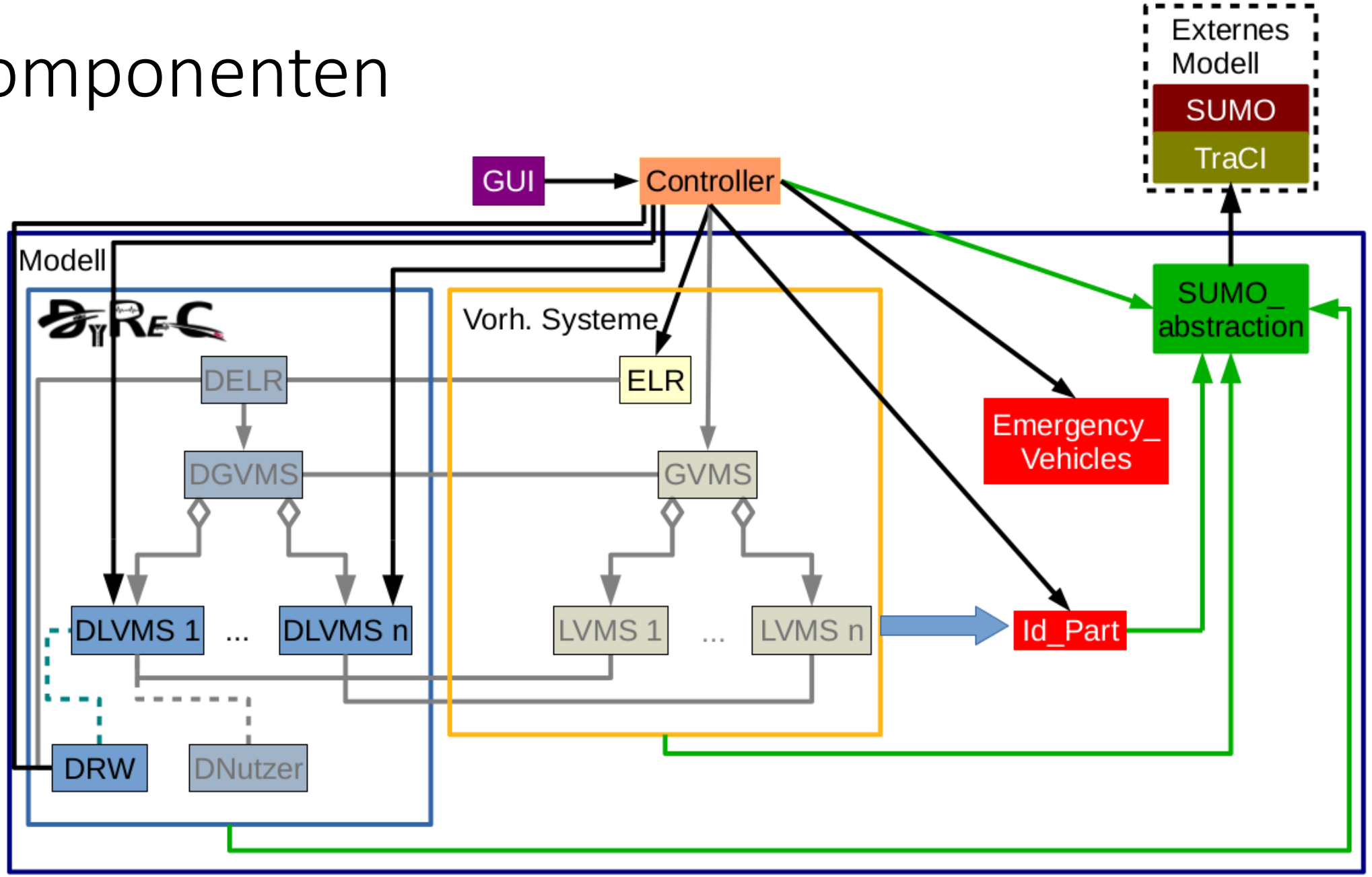
Komponenten



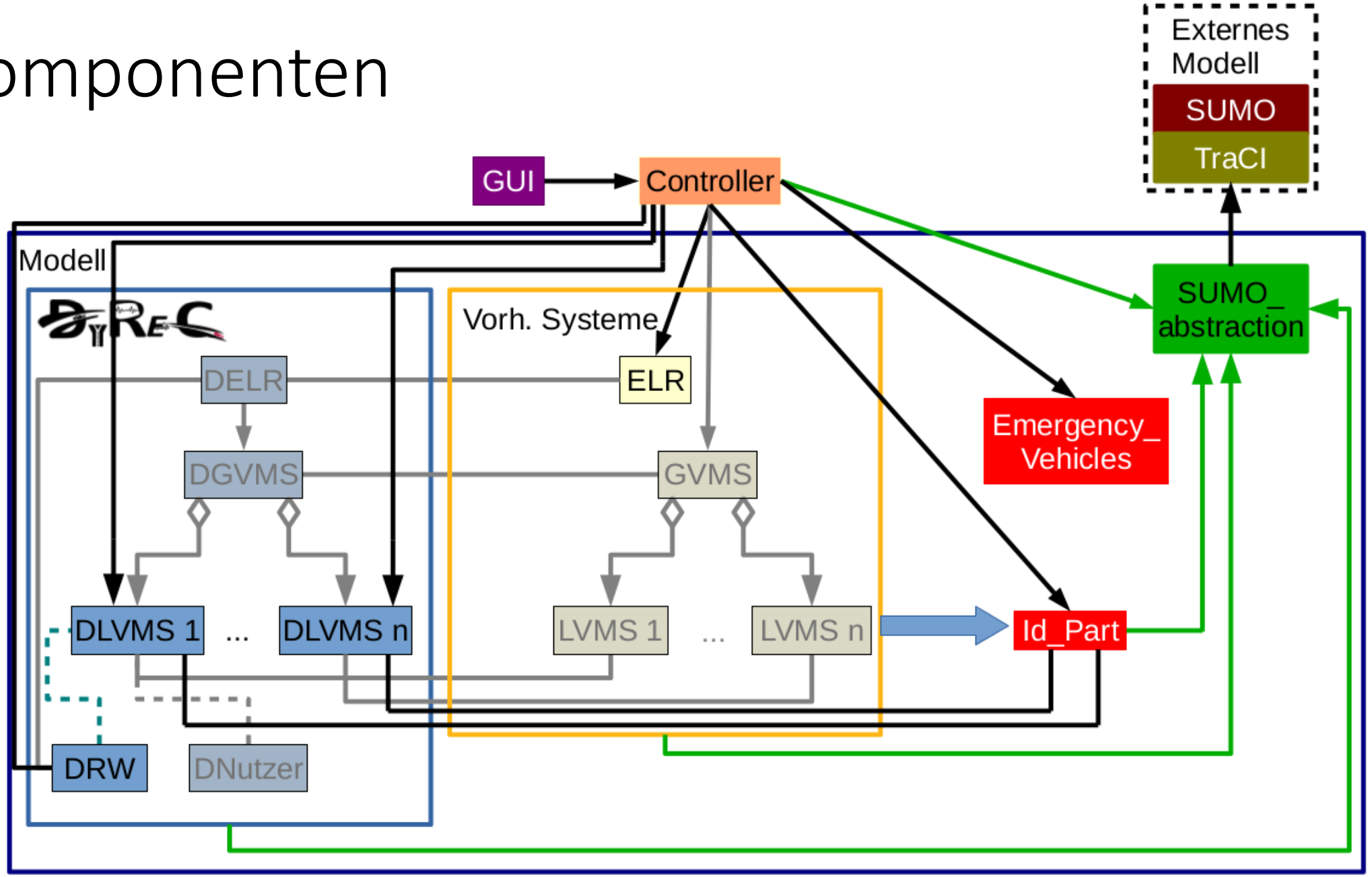
Komponenten



Komponenten



Komponenten

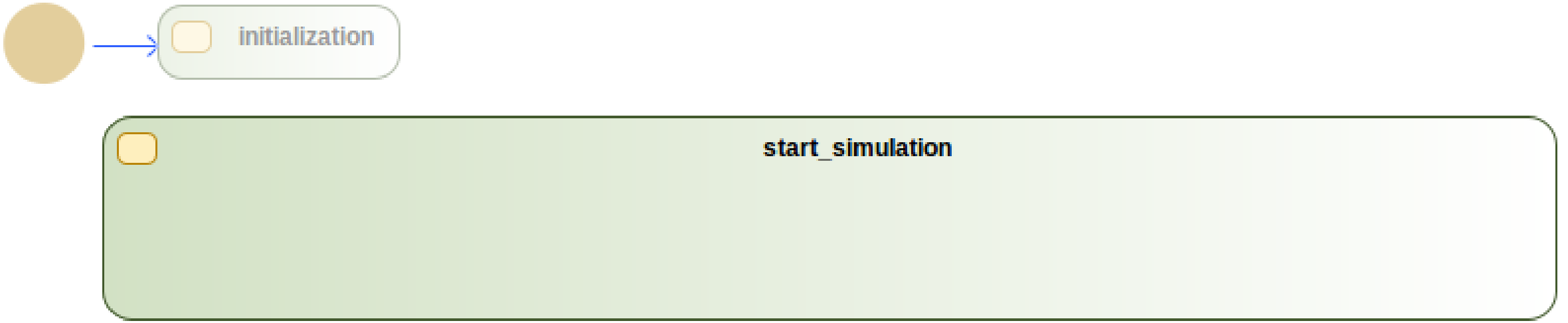


Agenda

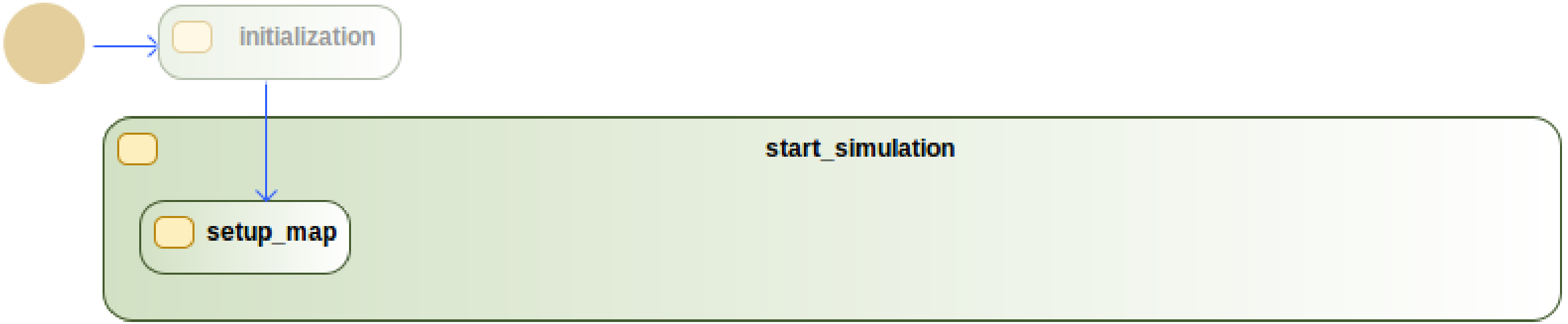
- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
  - SUMO-Umgebung
  - Komponentendiagramm
  - Programmablauf
  - Kaskadierendes Verfahren
  - LSA-Anpassung
  - Dynamische Rettungsgasse
  - Tests
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Programmablauf

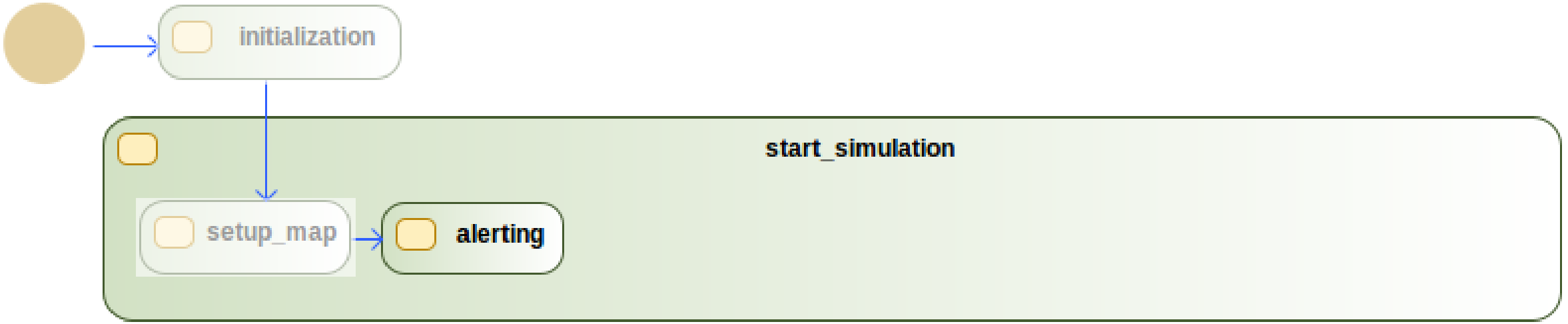
Programmablauf



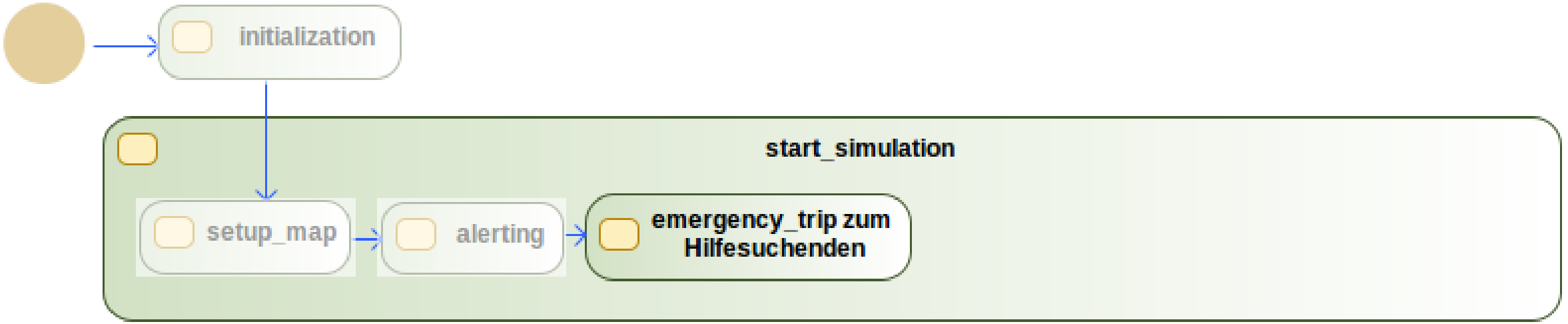
Programmablauf



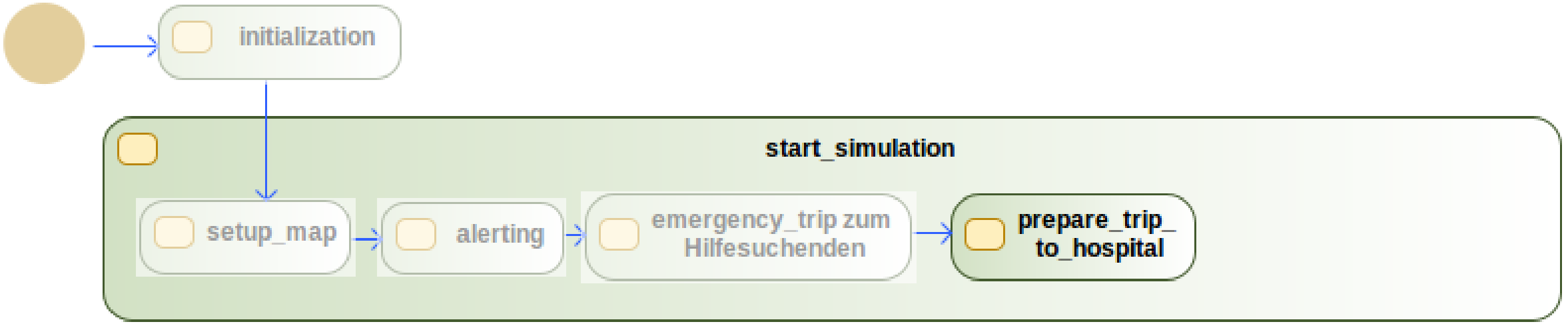
Programmablauf



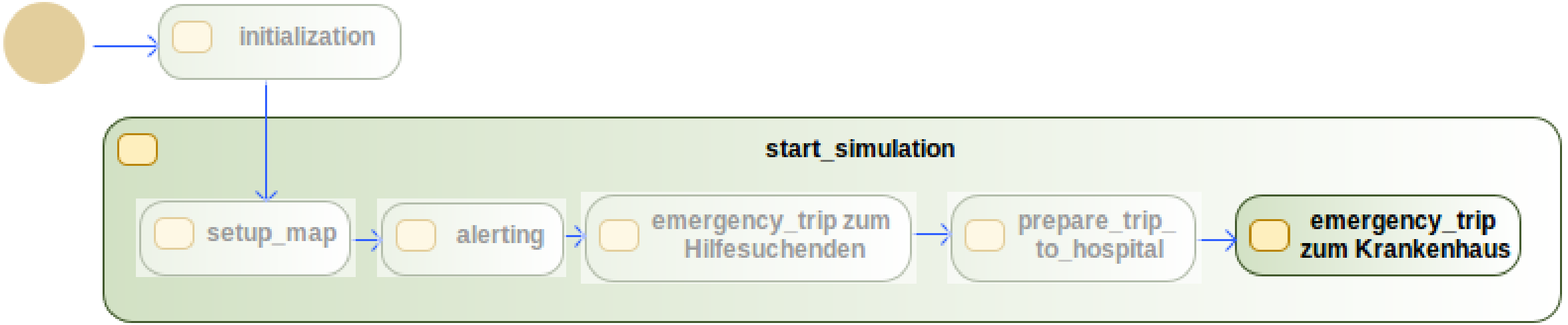
Programmablauf



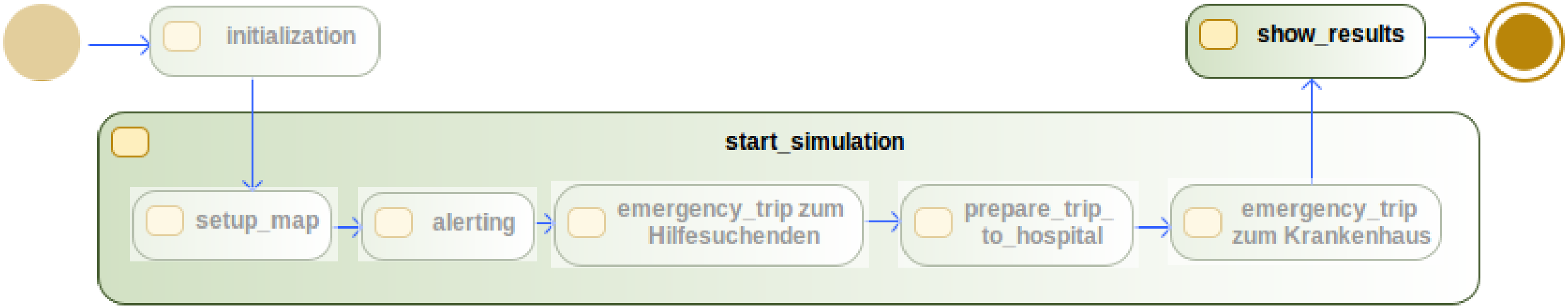
Programmablauf



Programmablauf



Programmablauf

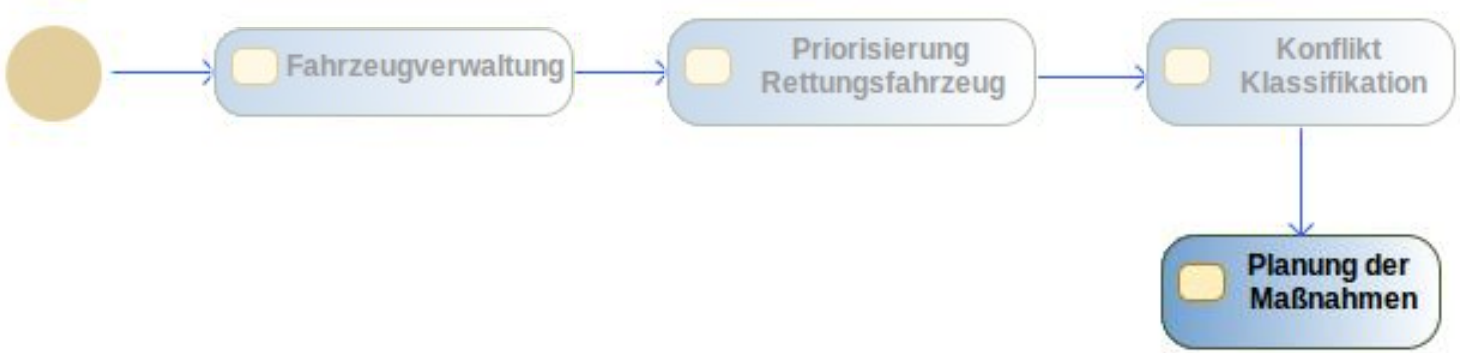


Programmablauf auf DLVMS-Ebene

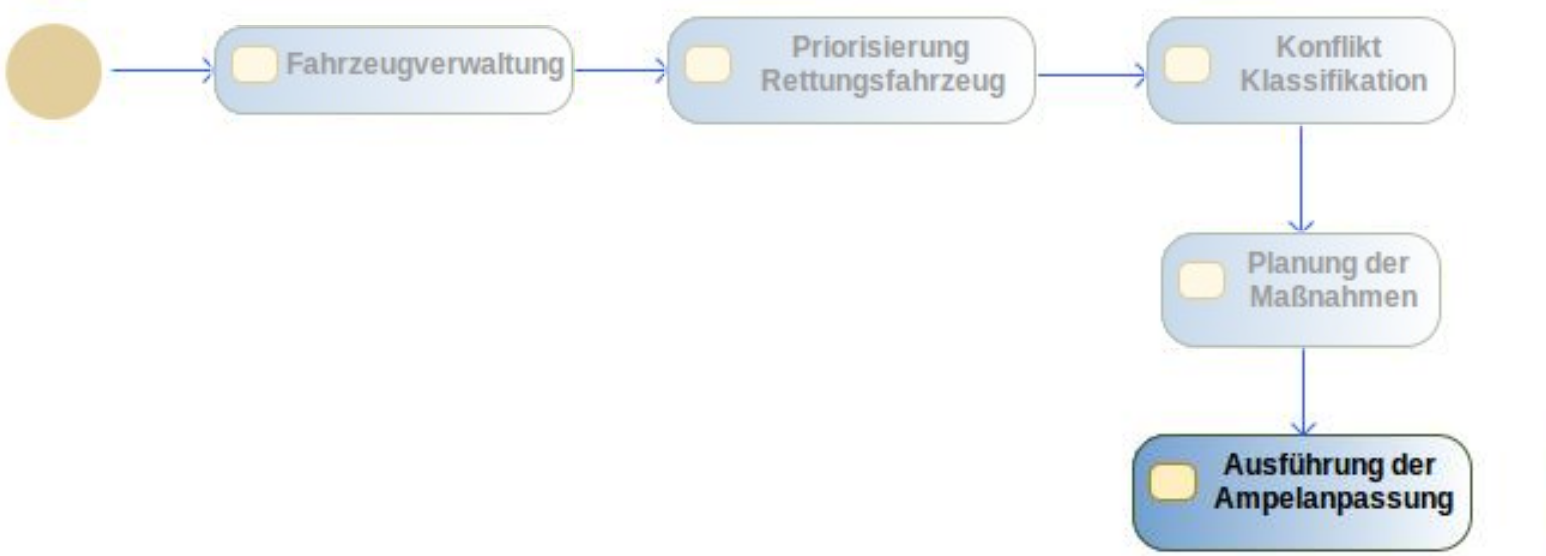
Programmablauf auf DLVMS-Ebene

Programmablauf auf DLVMS-Ebene

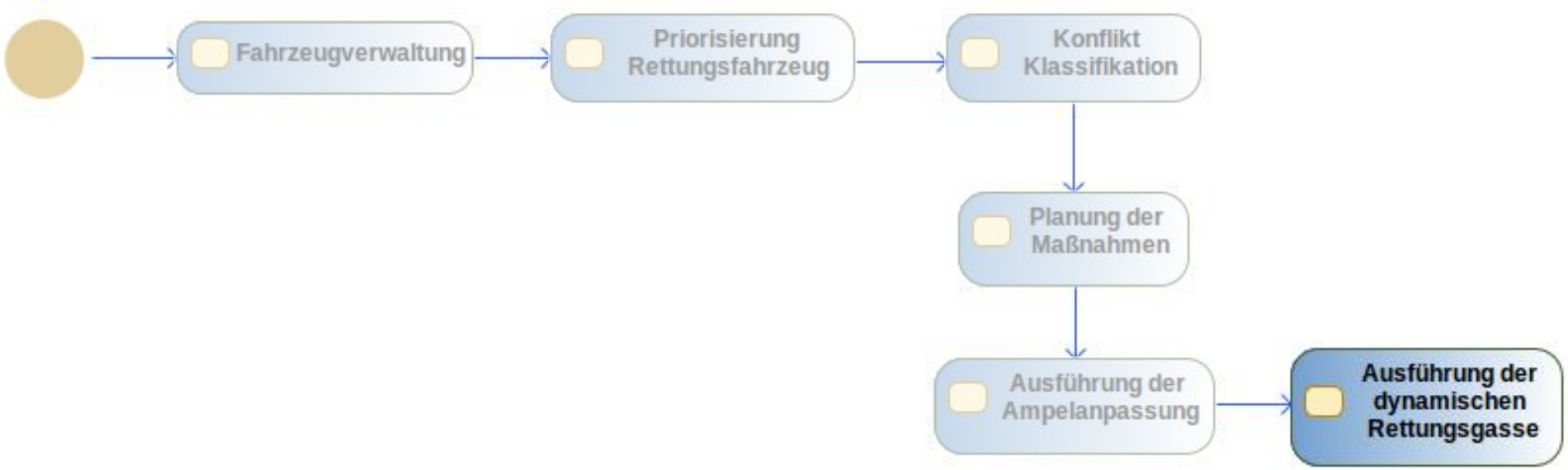
Programmablauf auf DLVMS-Ebene



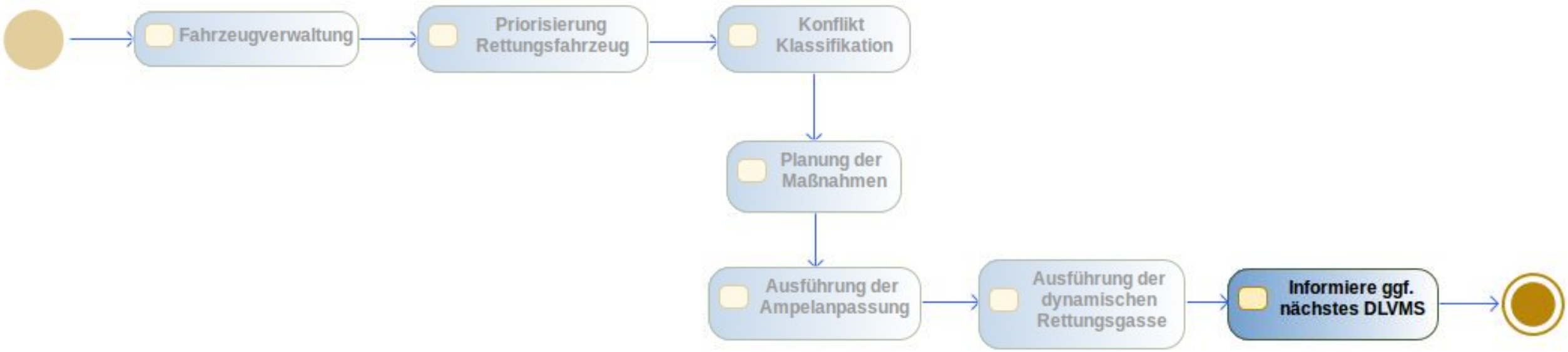
Programmablauf auf DLVMS-Ebene



Programmablauf auf DLVMS-Ebene



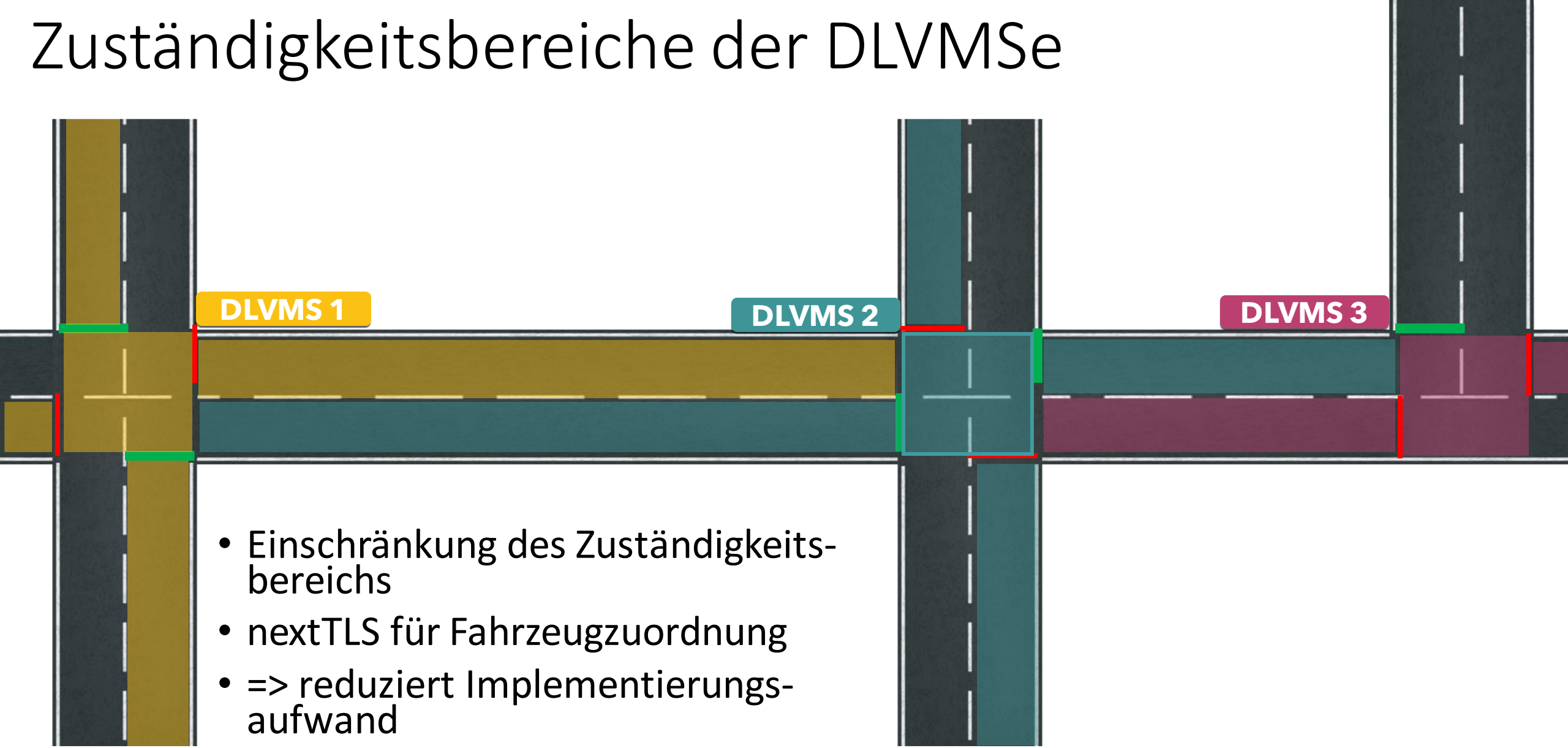
Programmablauf auf DLVMS-Ebene



Agenda

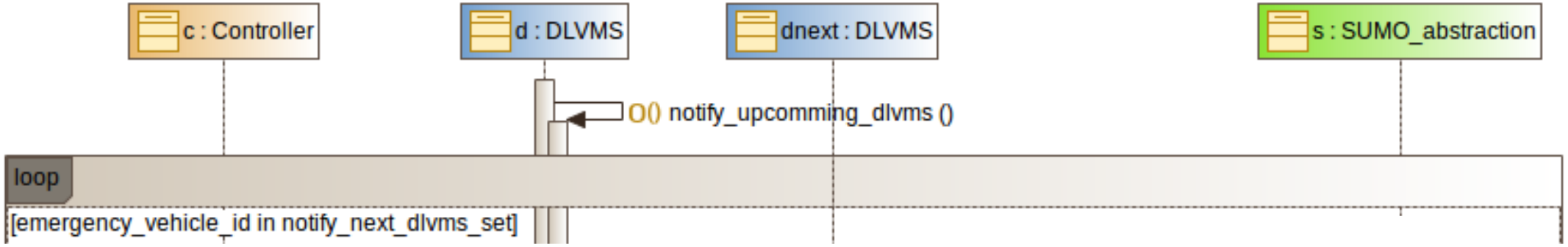
- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
  - SUMO-Umgebung
  - Komponentendiagramm
  - Programmablauf
  - Kaskadierendes Verfahren
  - LSA-Anpassung
  - Dynamische Rettungsgasse
  - Tests
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Zuständigkeitsbereiche der DLVMSe

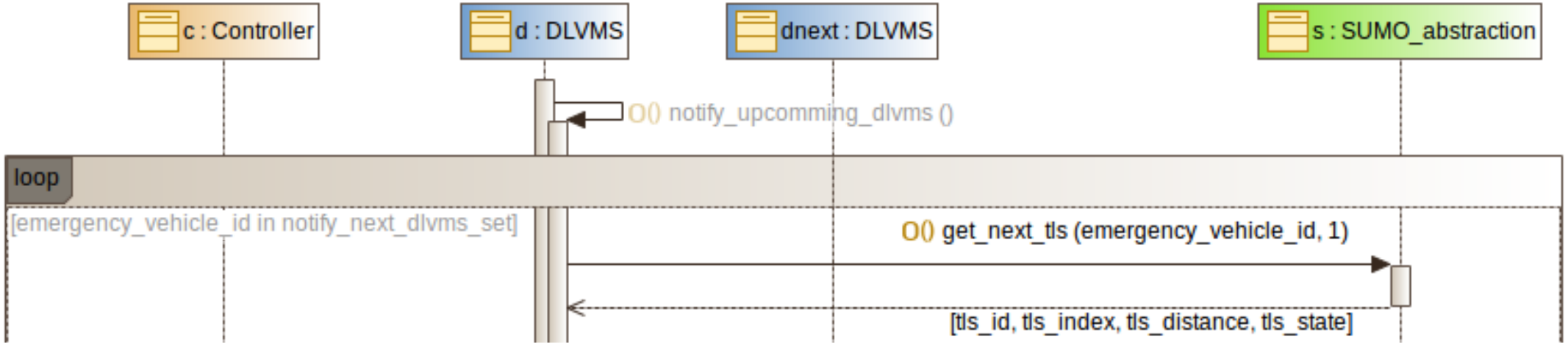


- Einschränkung des Zuständigkeits-bereichs
- nextTLS für Fahrzeugzuordnung
- => reduziert Implementierungs-aufwand
- Aber: Probleme am Ende der Fahrt eines Fahrzeugs

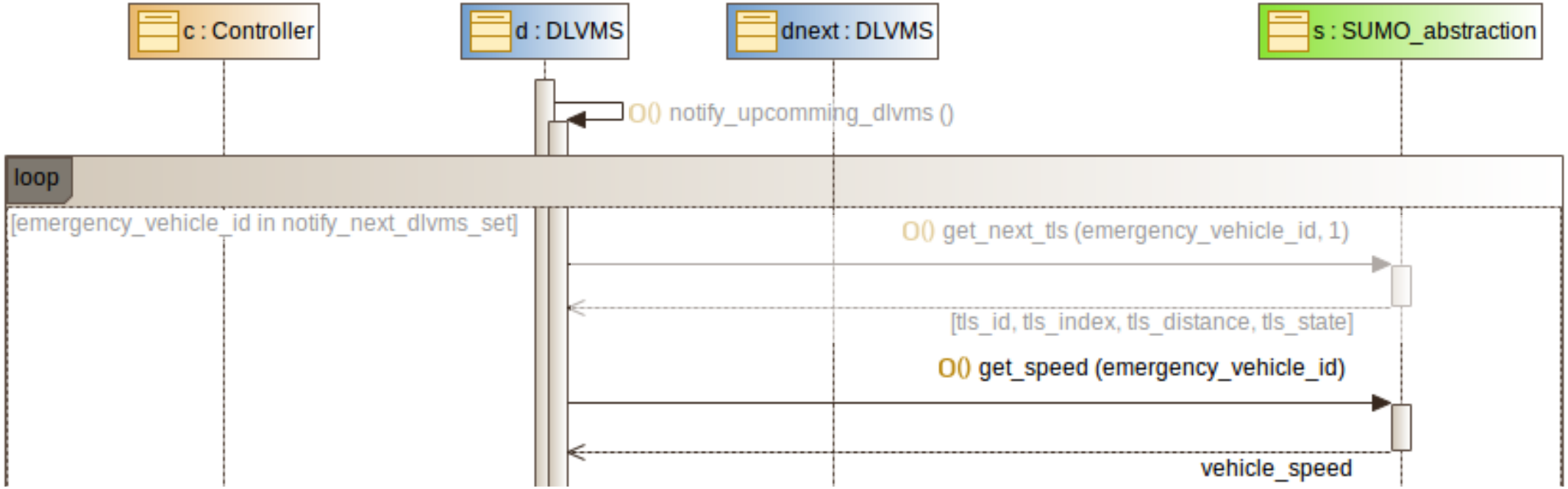
Kaskadierendes Verfahren



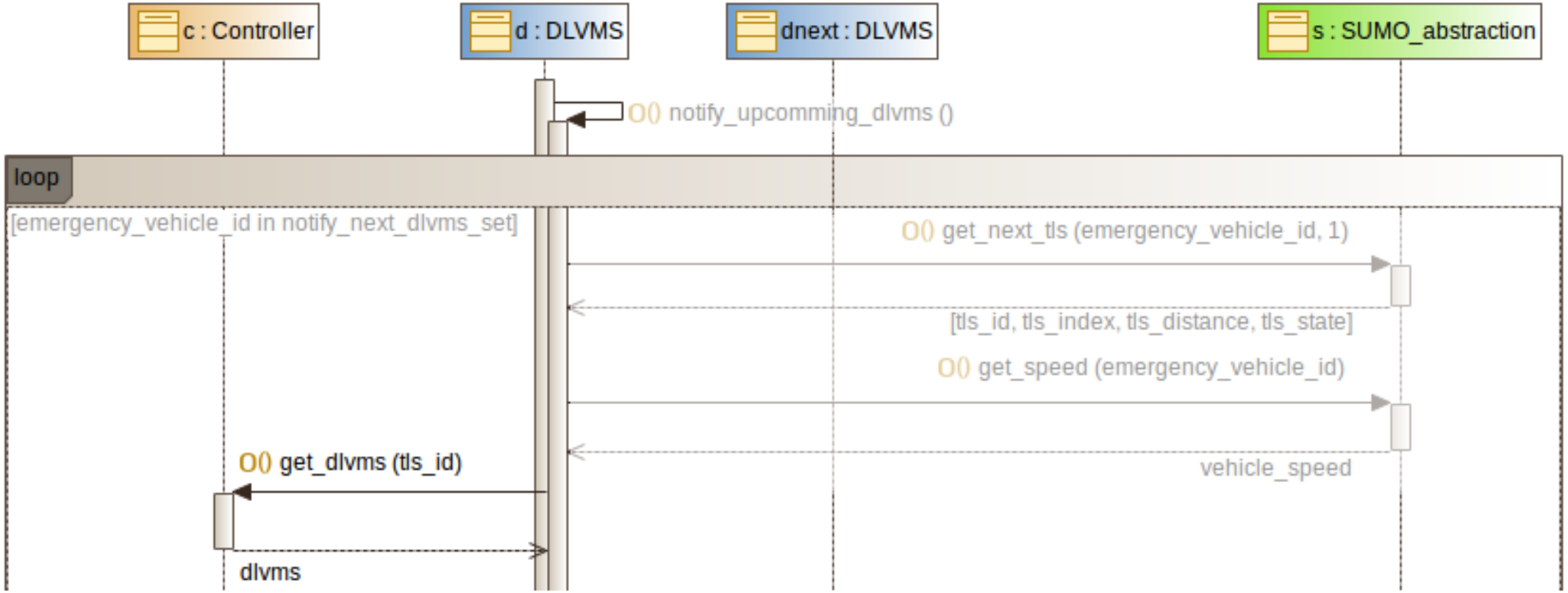
Kaskadierendes Verfahren



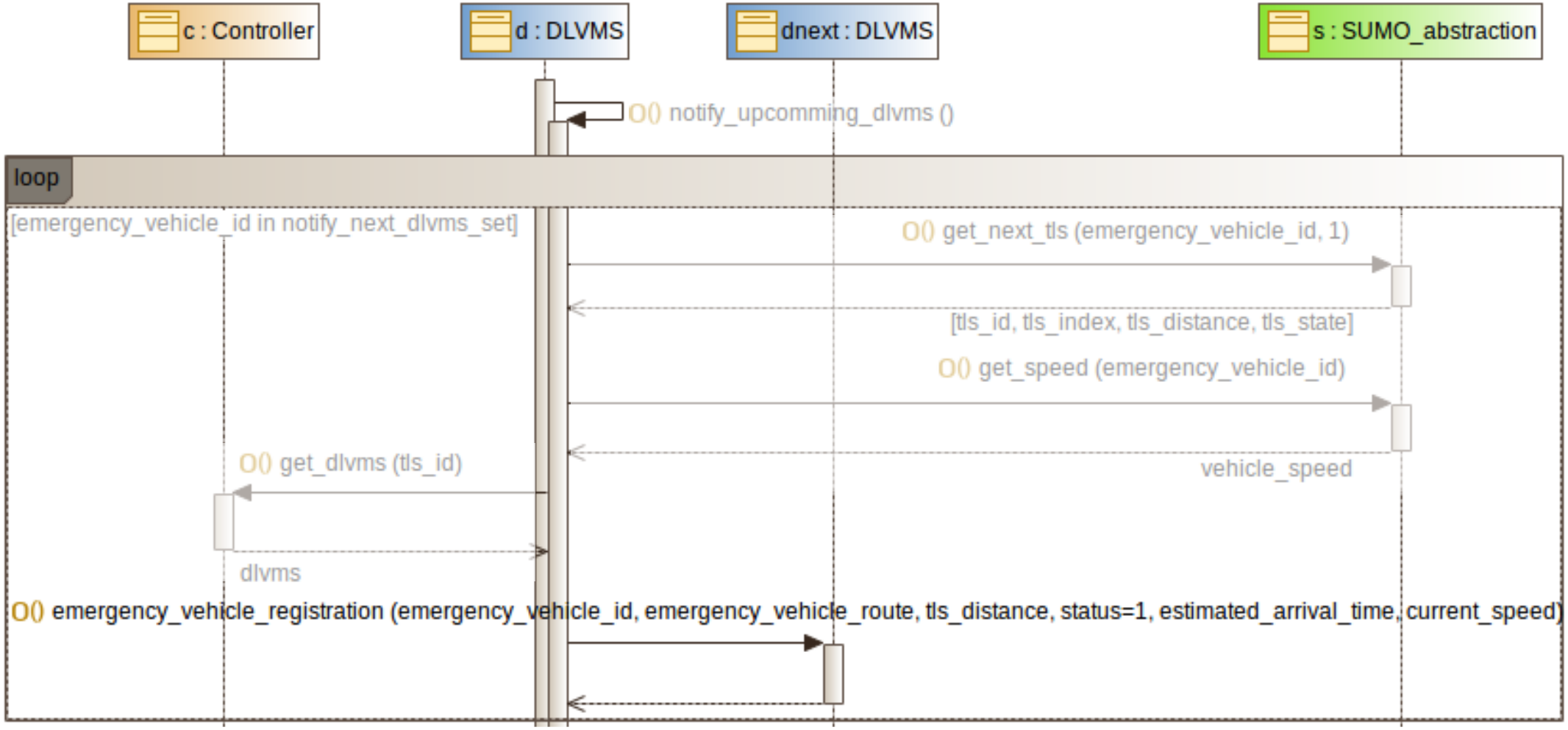
Kaskadierendes Verfahren



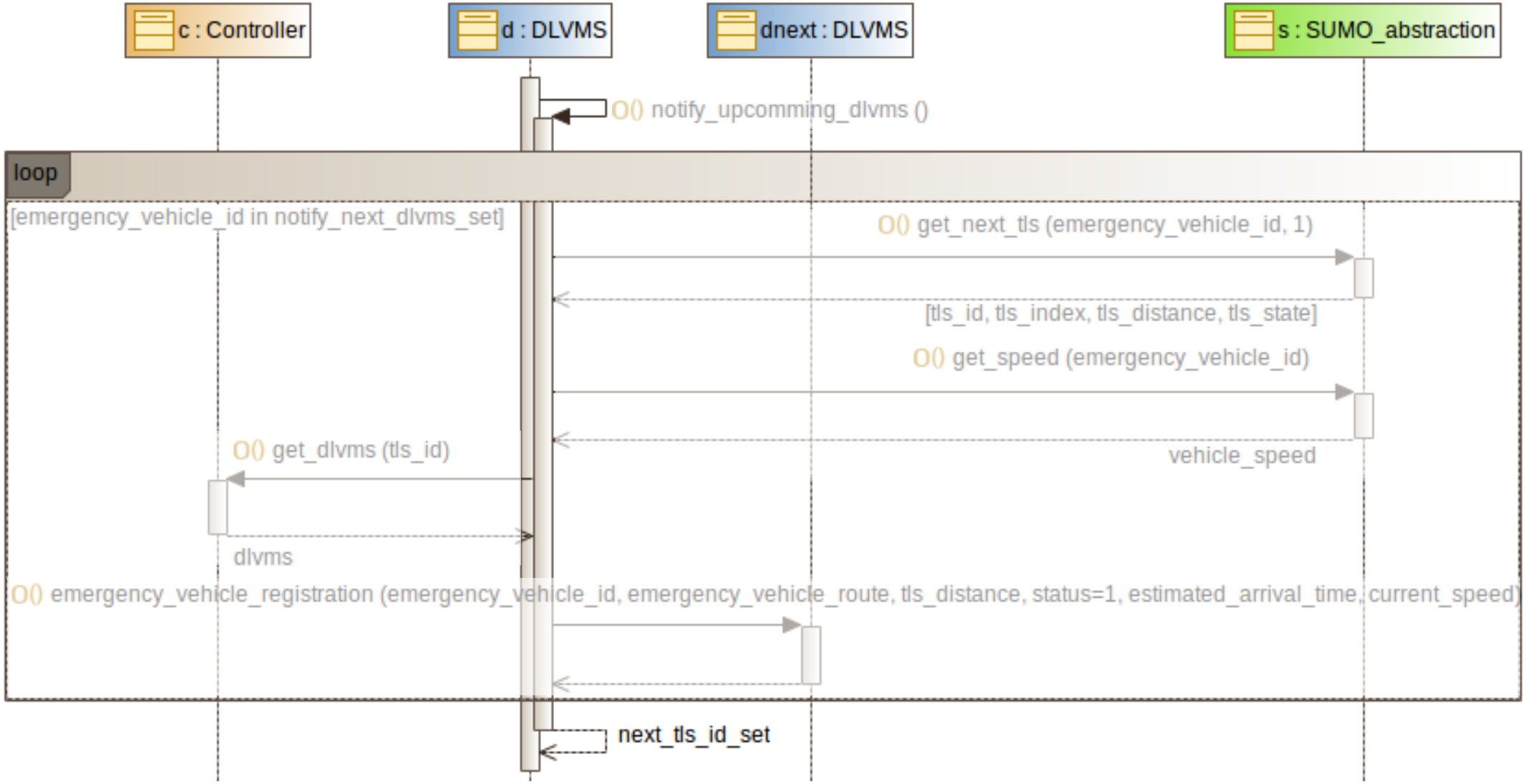
Kaskadierendes Verfahren



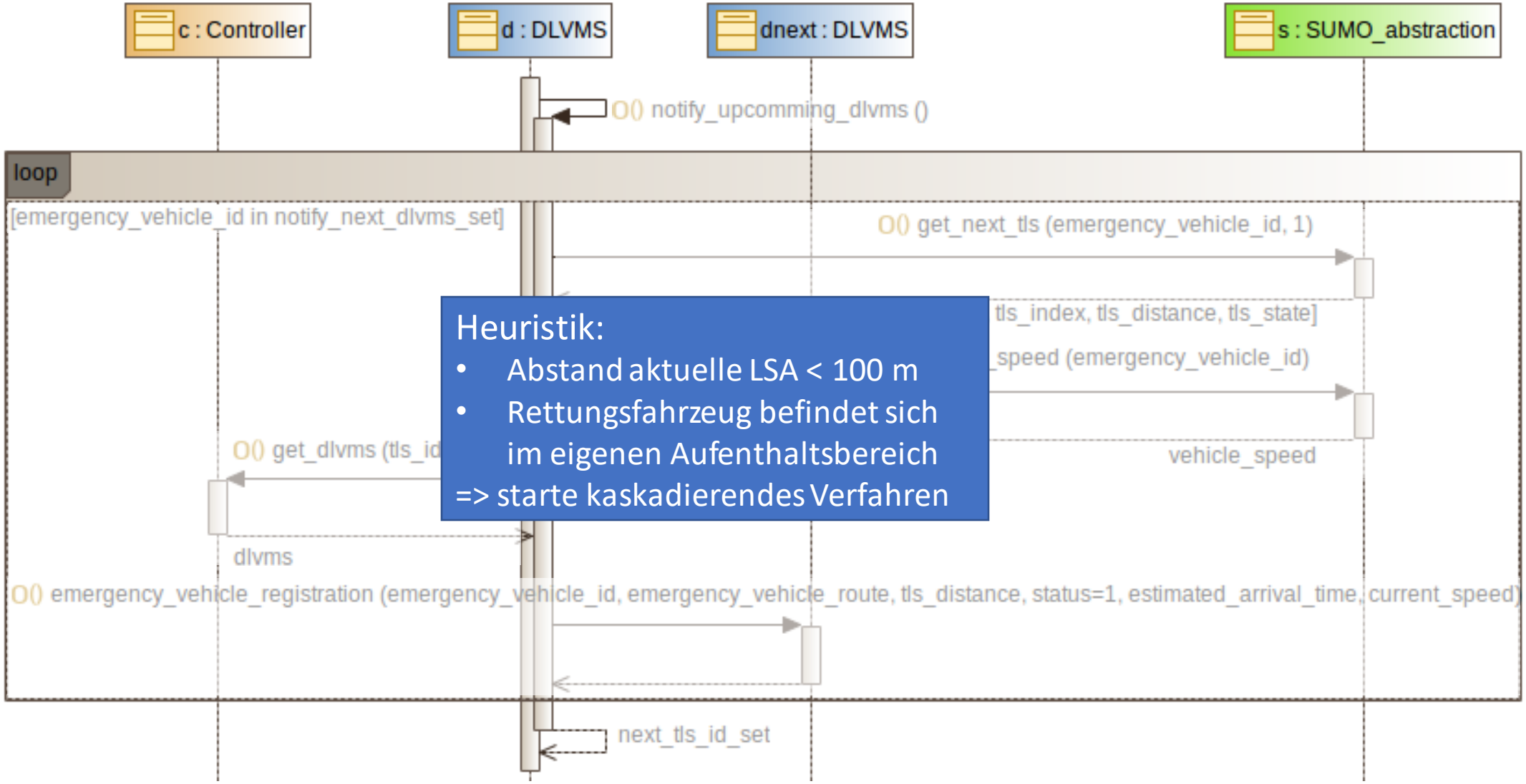
Kaskadierendes Verfahren



Kaskadierendes Verfahren



Kaskadierendes Verfahren



Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
  - SUMO-Umgebung
  - Komponentendiagramm
  - Programmablauf
  - Kaskadierendes Verfahren
  - LSA-Anpassung
  - Dynamische Rettungsgasse
  - Tests
- Demonstrator
- Evaluation
- Ausblick
- Fazit

LSA-Anpassung

- Ursprüngliche Idee:
  - Überführe LSA in Notfallprogramm, sodass das Rettungsfahrzeug über eine grüne Ampel fährt
  - Umschaltvorgänge sollen sicher sein
  - => Automat aus Konzept umsetzen
  - Bei Zustandsübergang neues Programm in SUMO
- Problem: Die Übergangszeiten werden nicht gehalten
- Daher nicht vollständig im Demonstrator umgesetzt

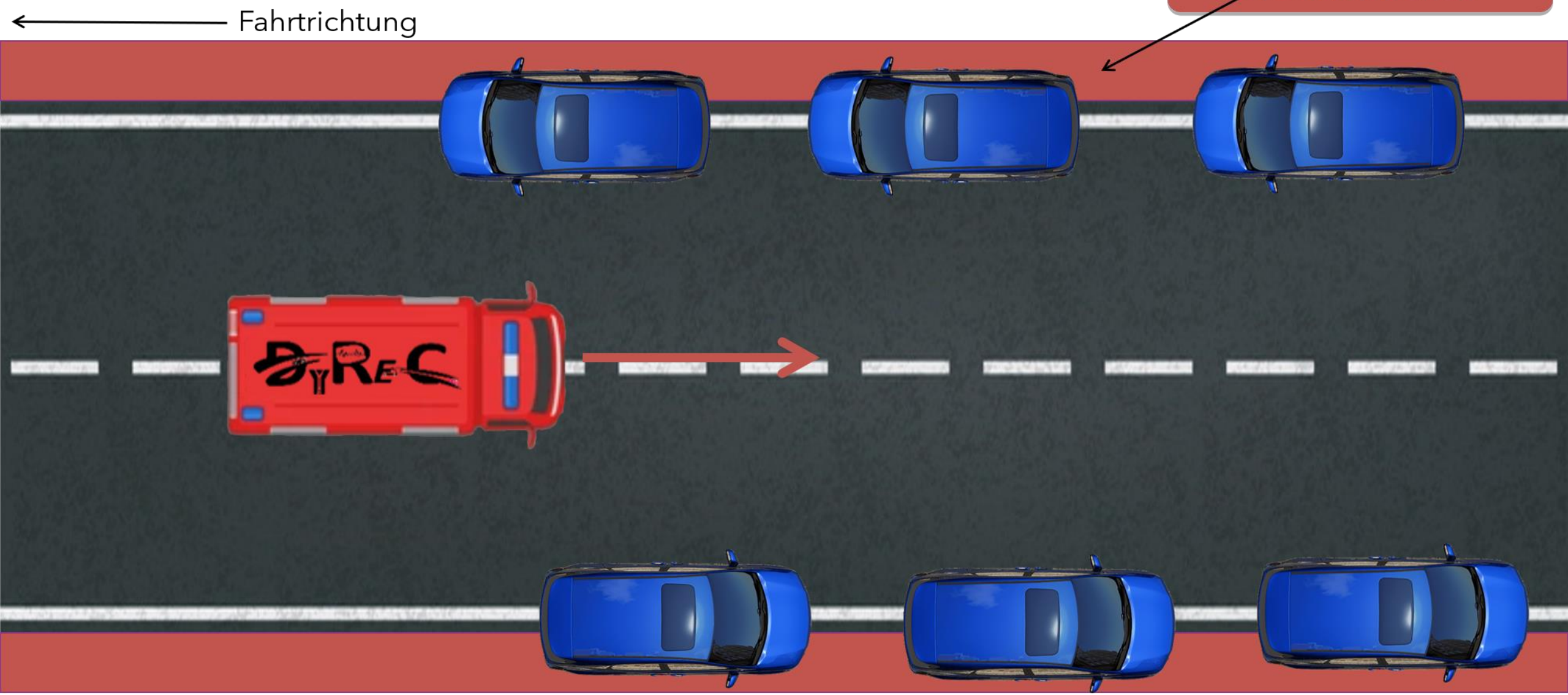
LSA-Anpassung

- Ursprüngliche Idee:
  - Überführe LSA in Notfallprogramm, sodass das Rettungsfahrzeug über eine grüne Ampel fährt
  - Umschaltvorgänge sollen sicher sein
  - => Automat aus Konzept umsetzen
  - Bei Zustandsübergang neues Programm in SUMO
- Problem: Die Übergangszeiten werden nicht gehalten
- Daher nicht vollständig im Demonstrator umgesetzt

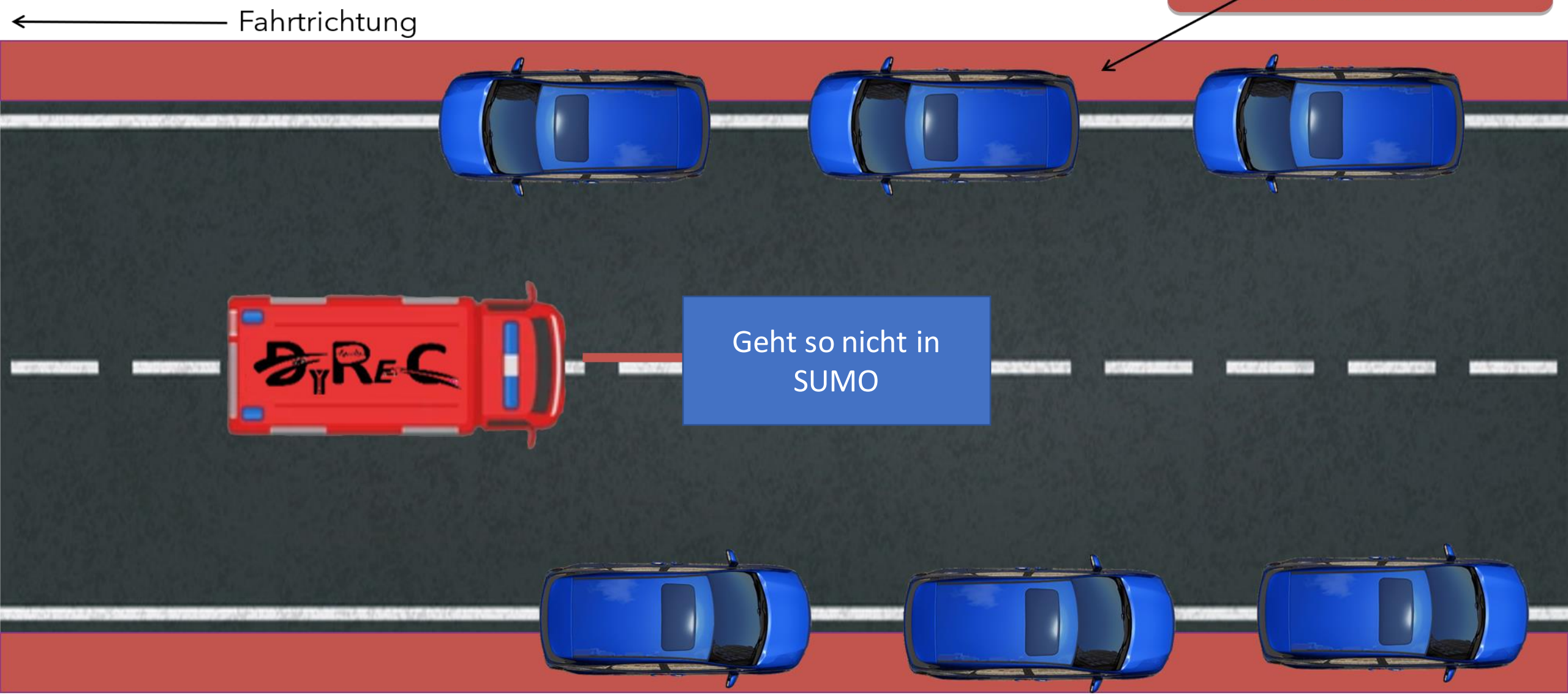
Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
  - SUMO-Umgebung
  - Komponentendiagramm
  - Programmablauf
  - Kaskadierendes Verfahren
  - LSA-Anpassung
  - Dynamische Rettungsgasse
  - Tests
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Dynamische Rettungsgasse in der Realität

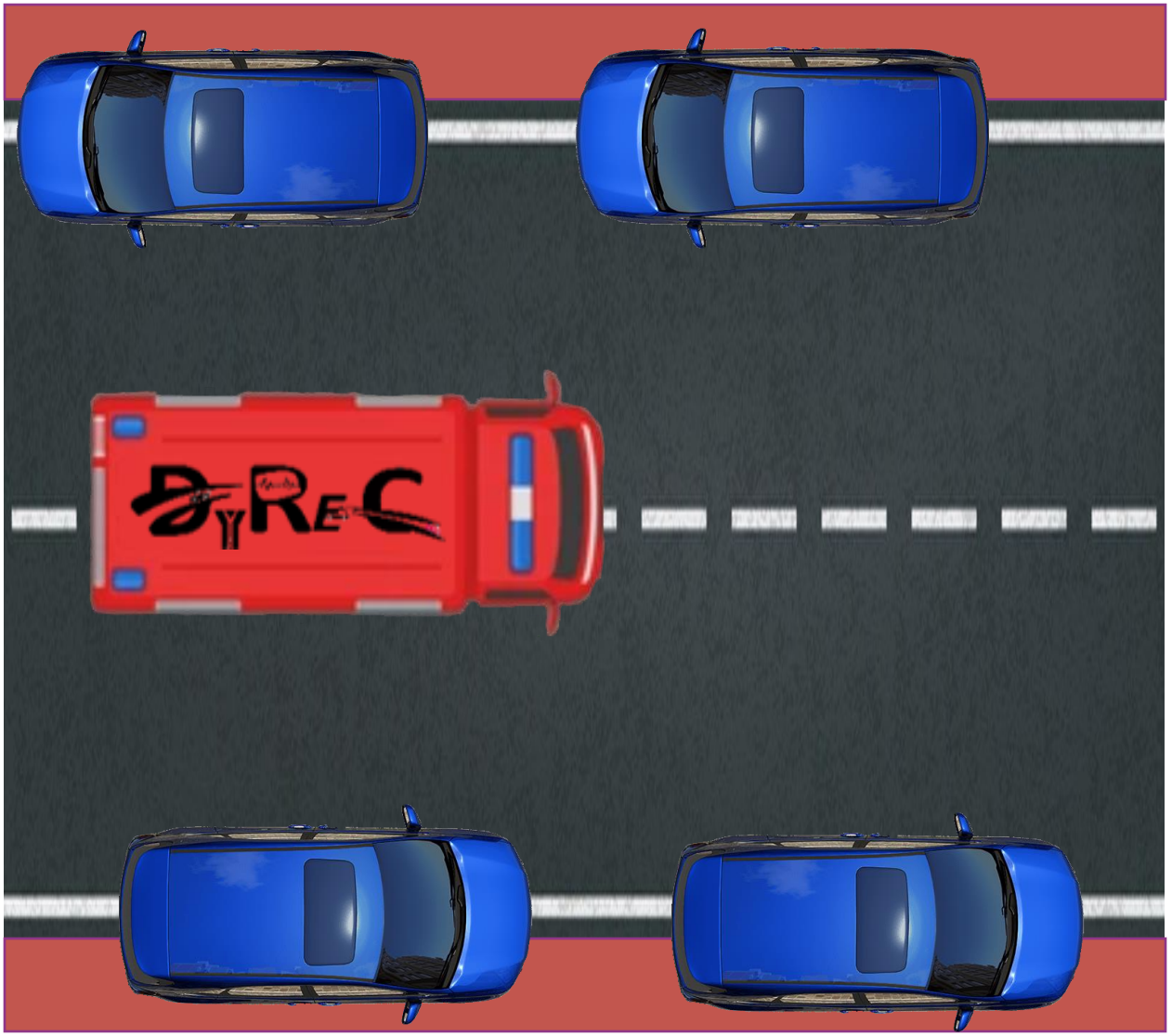


Dynamische Rettungsgasse in der Realität

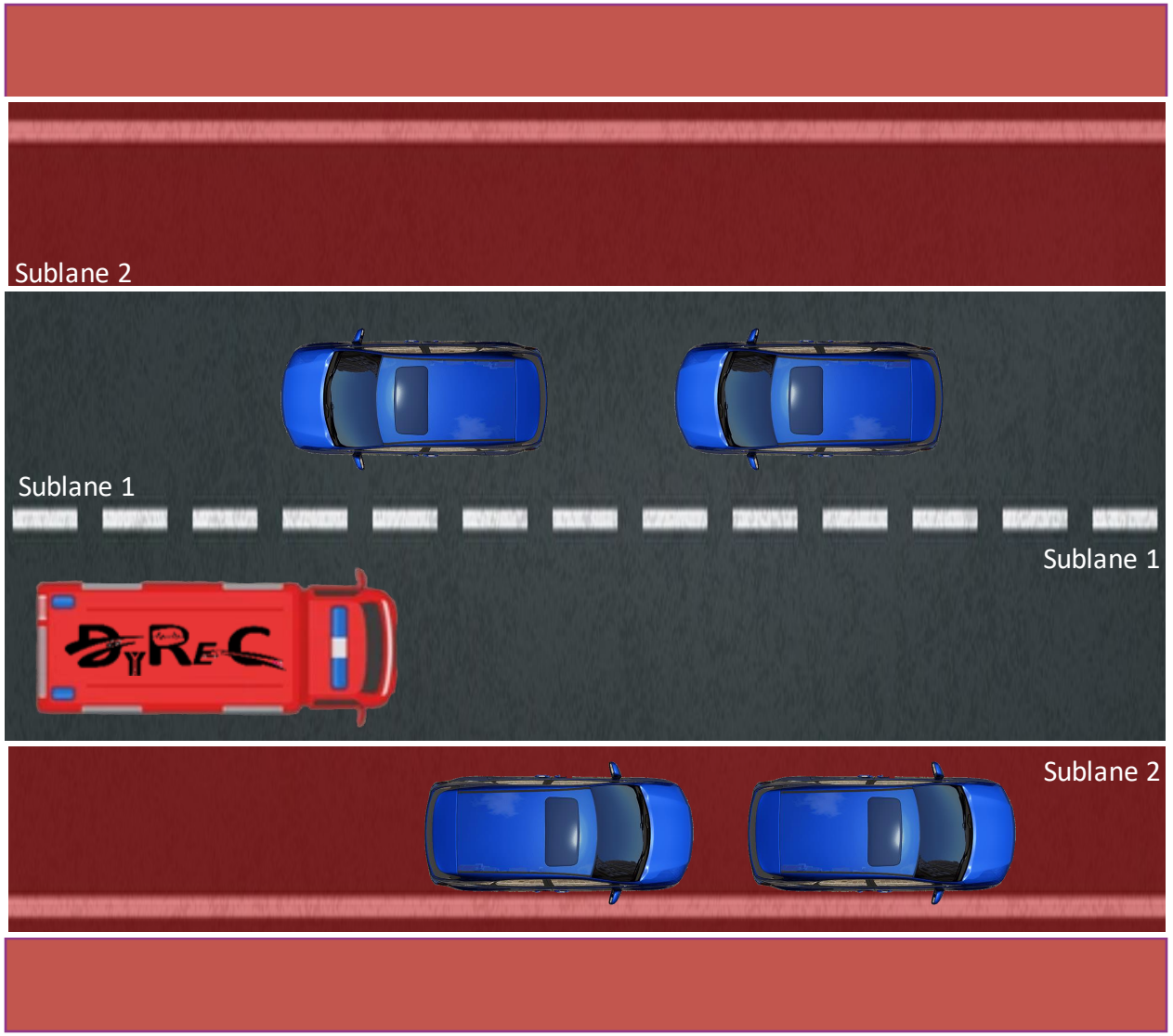


Dynamische Rettungsgasse

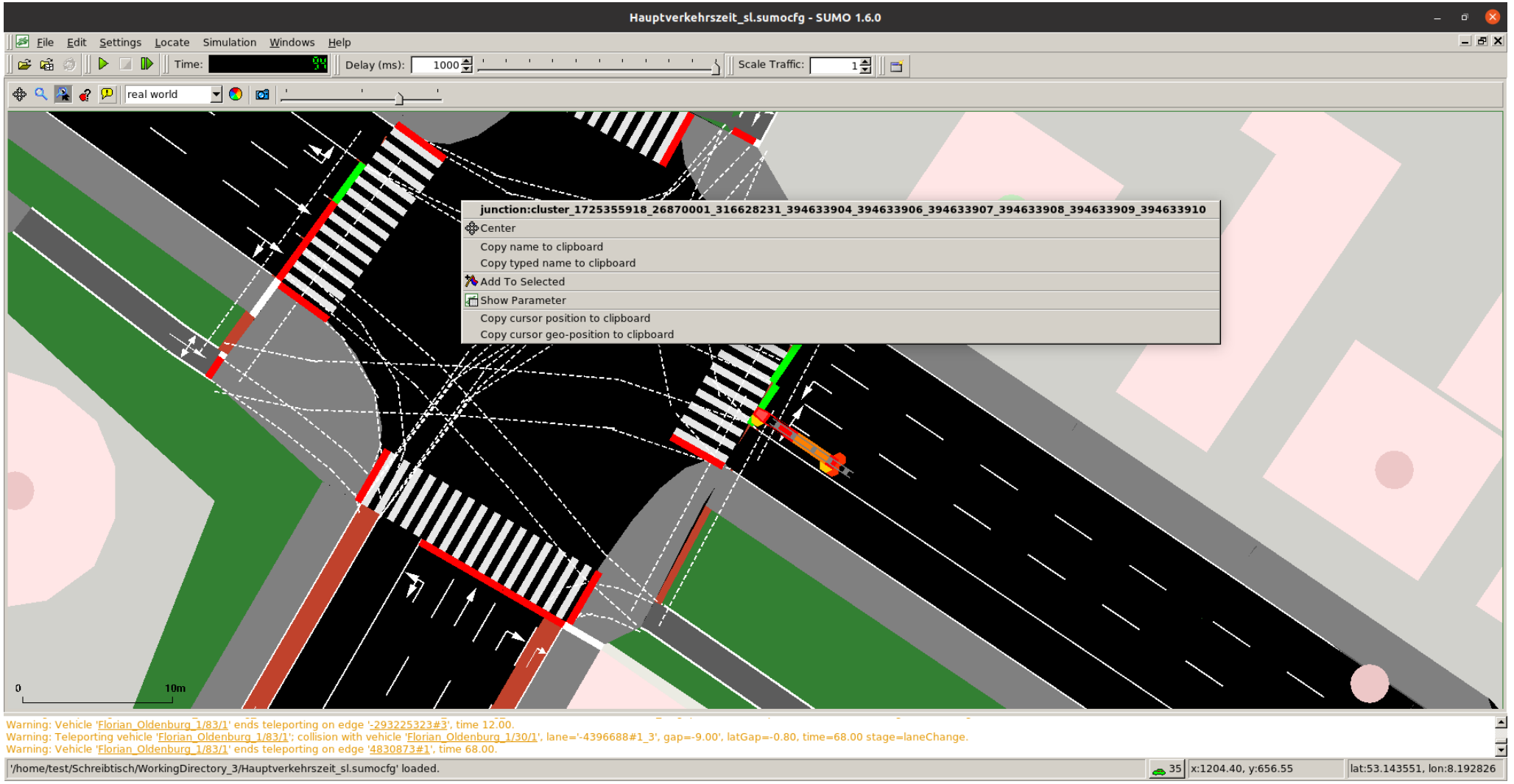
Realität



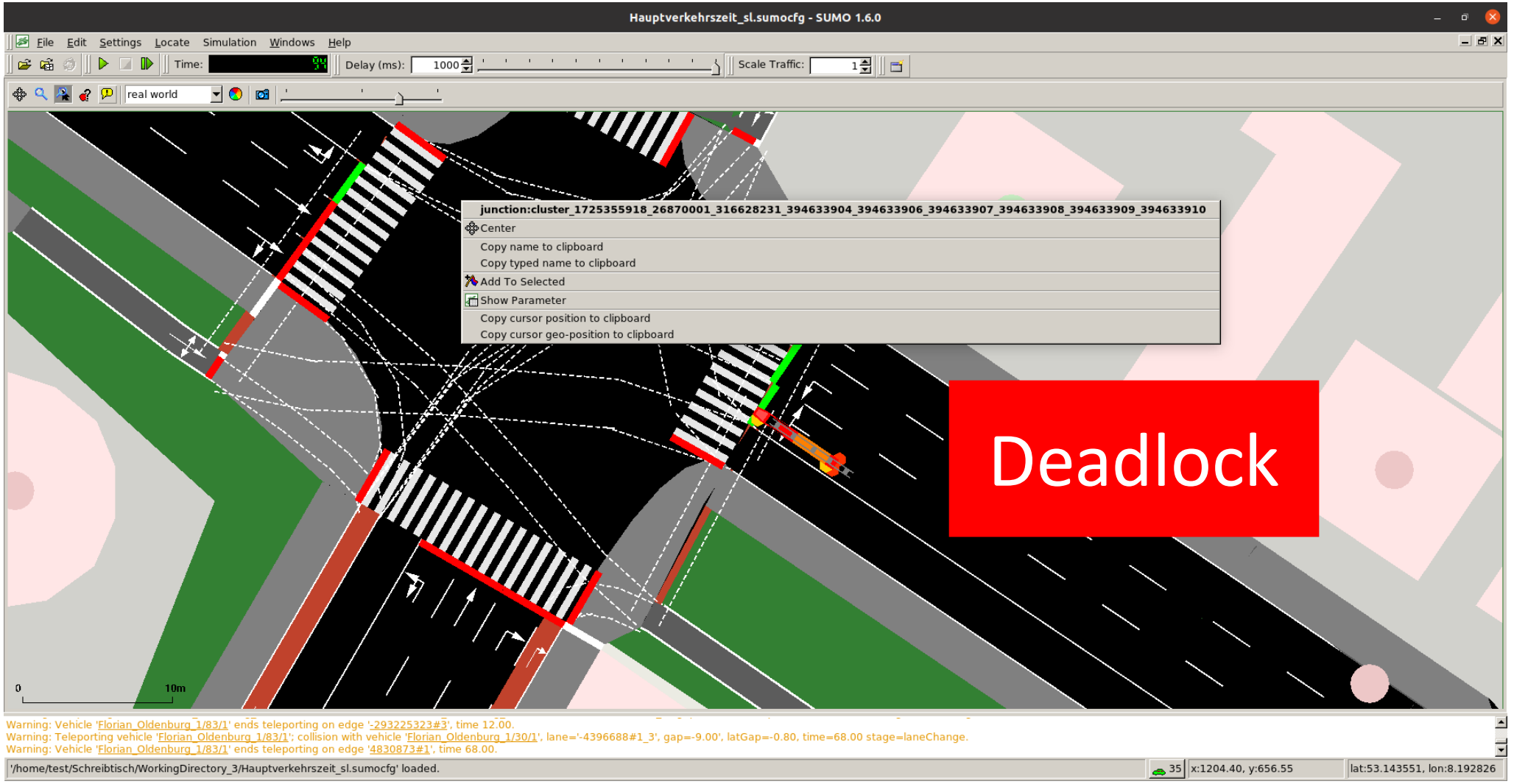
SUMO-Sublane Modell



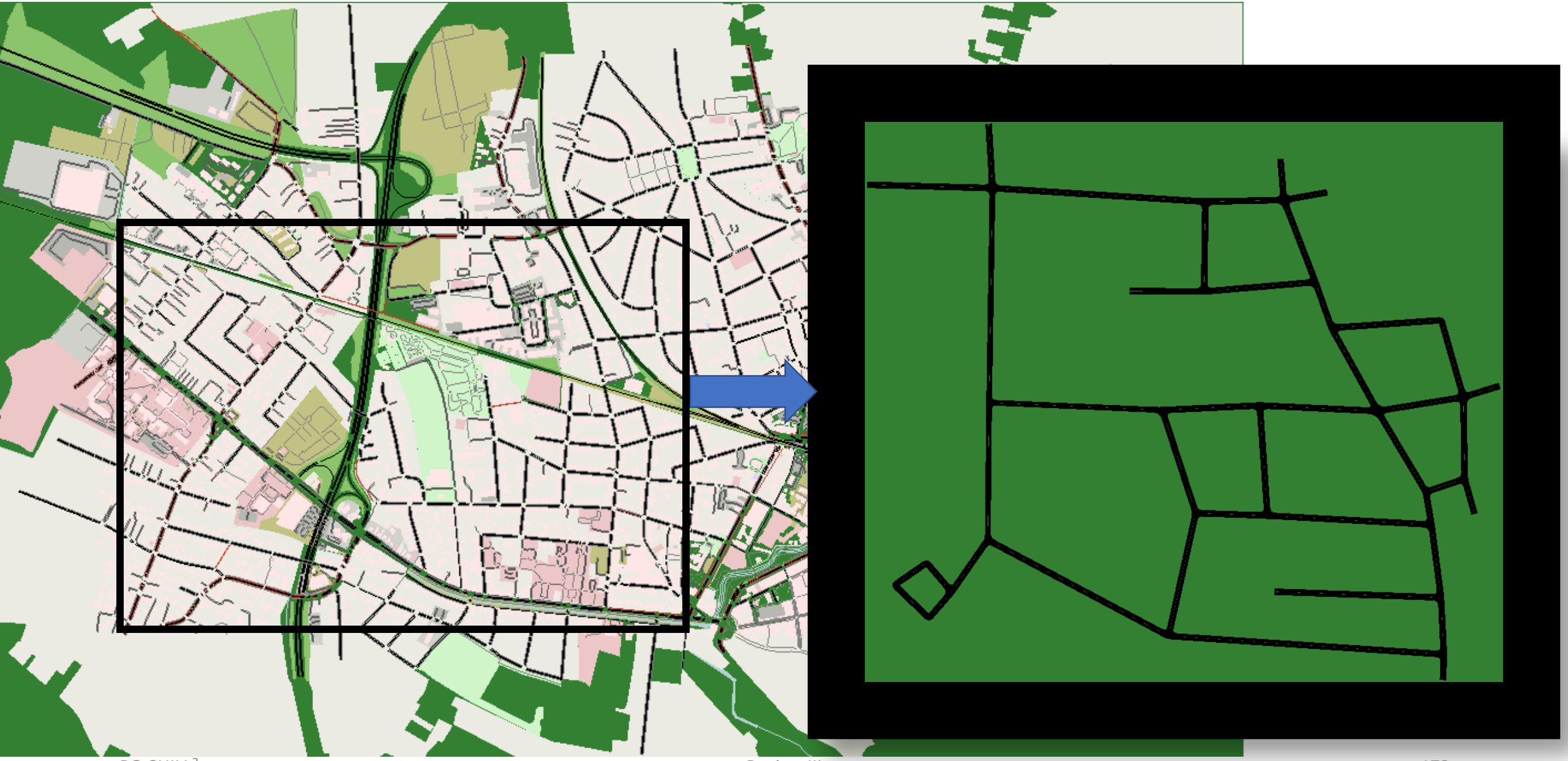
Probleme mit der Basiskarte …



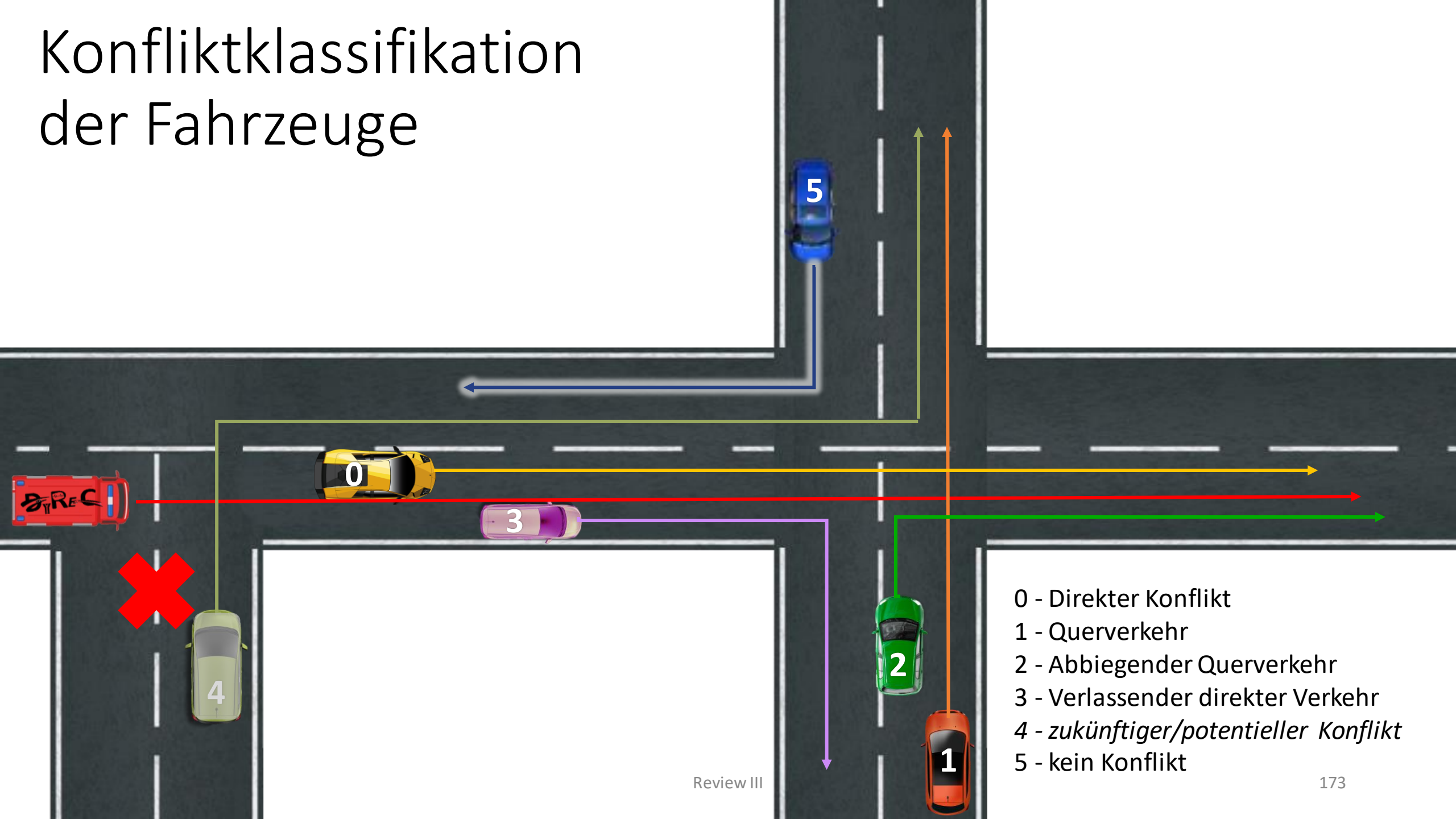
Probleme mit der Basiskarte …



Neudefinition der Basiskarte



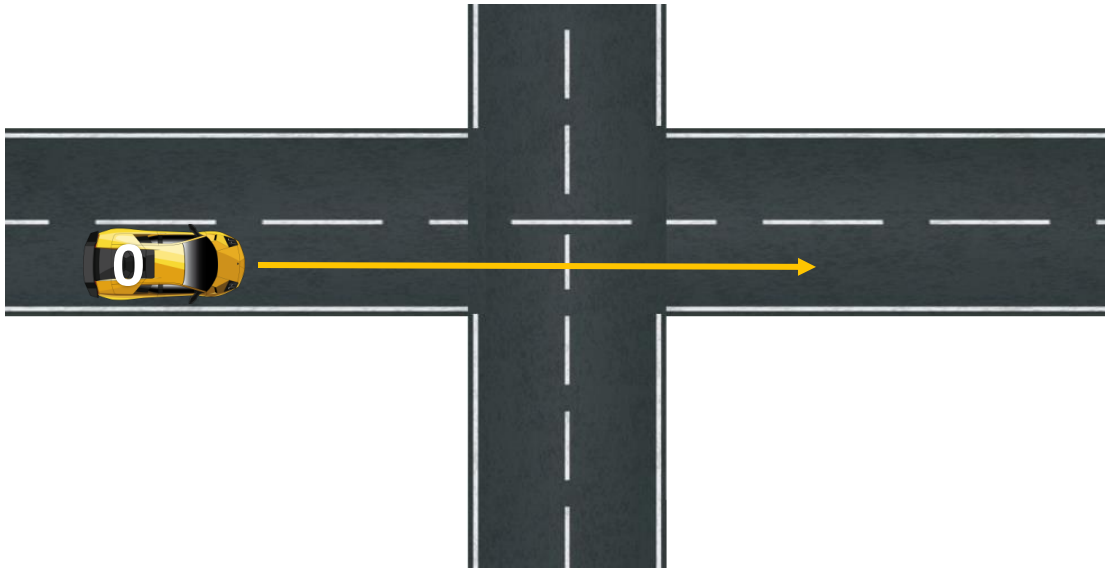
Konfliktklassifikation der Fahrzeuge

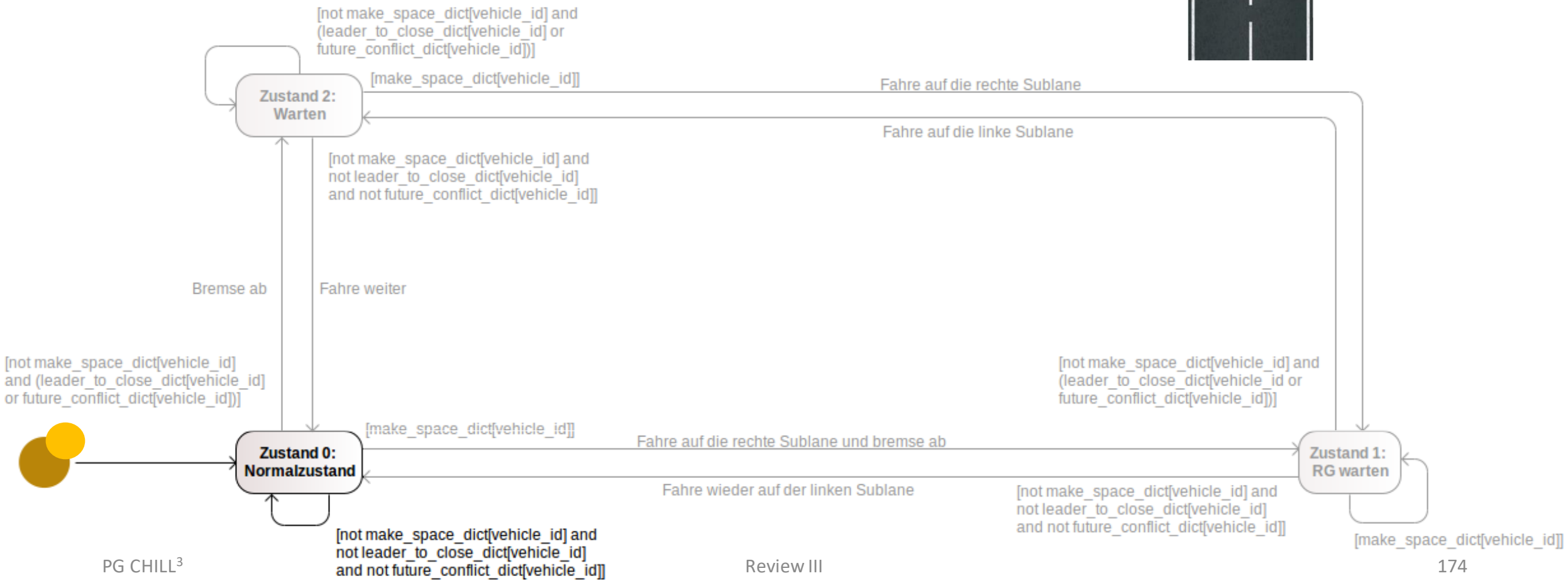


0 - Direkter Konflikt
1 - Querverkehr
2 - Abbiegender Querverkehr
3 - Verlassender direkter Verkehr
4 - zukünftiger/potentieller Konflikt
5 - kein Konflikt

Zustandsautomat eines Fahrzeugs

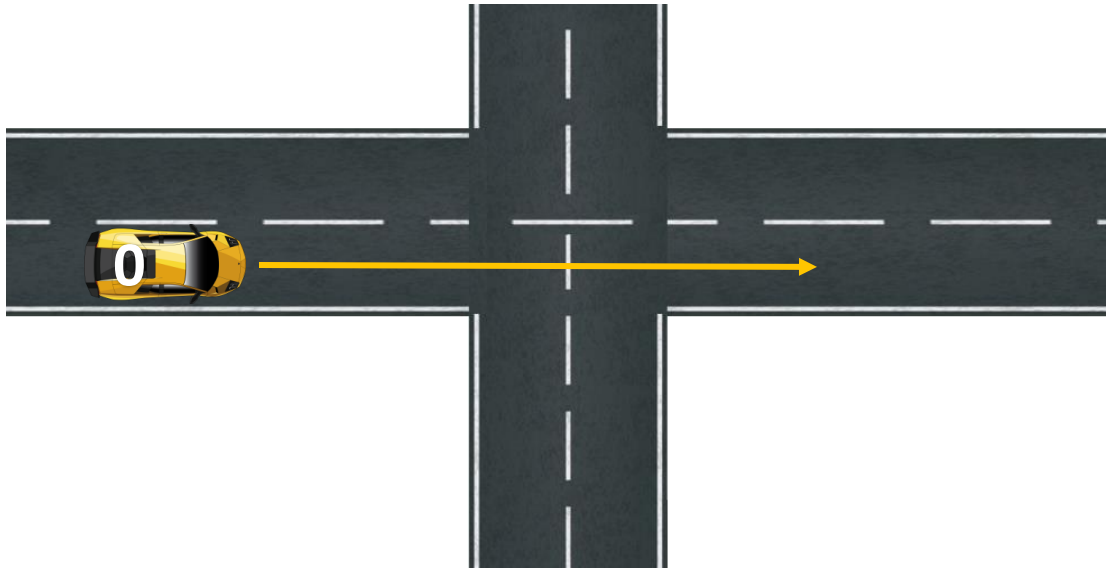
make_space_dict ={0: F}
leader_to_close_dict={0: F}
future_conflict_dict ={0: F}

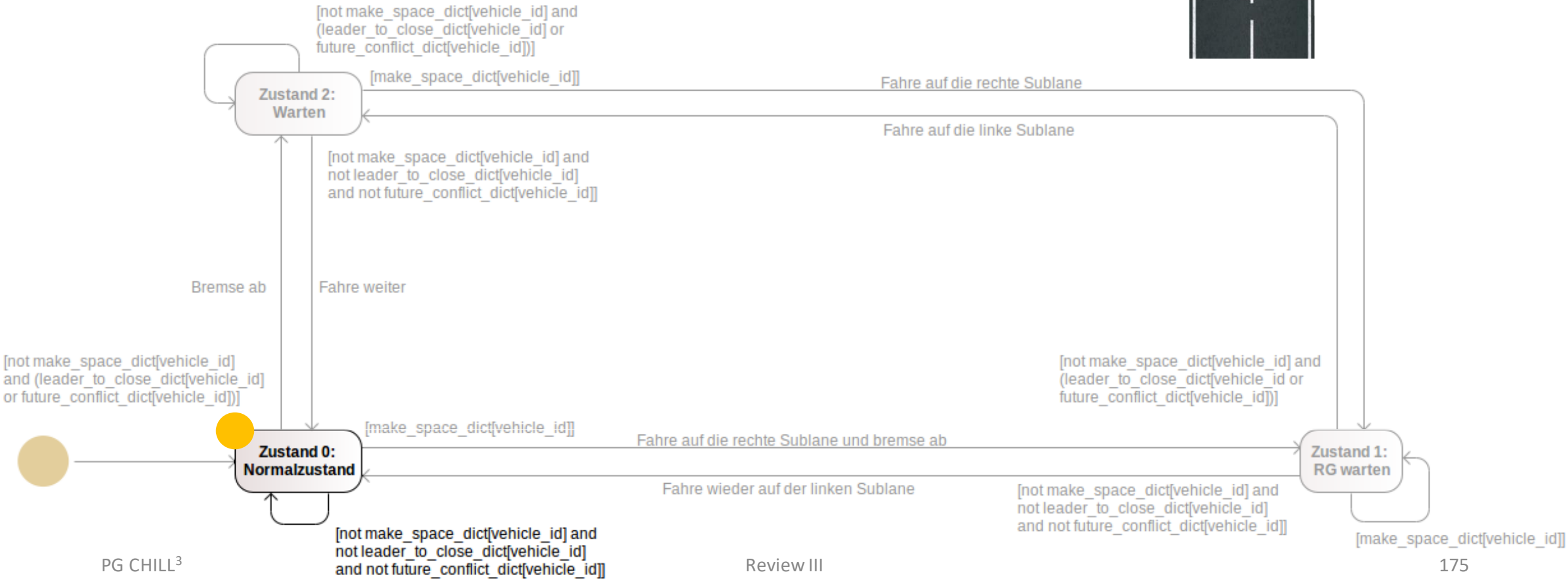




Zustandsautomat eines Fahrzeugs

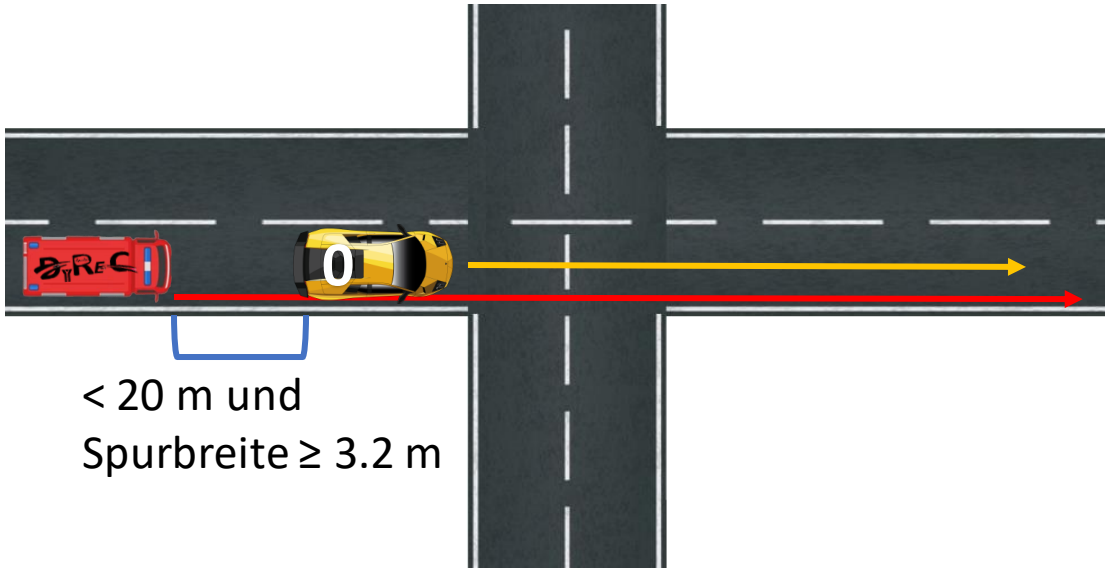
make_space_dict ={0: F}
leader_to_close_dict={0: F}
future_conflict_dict ={0: F}

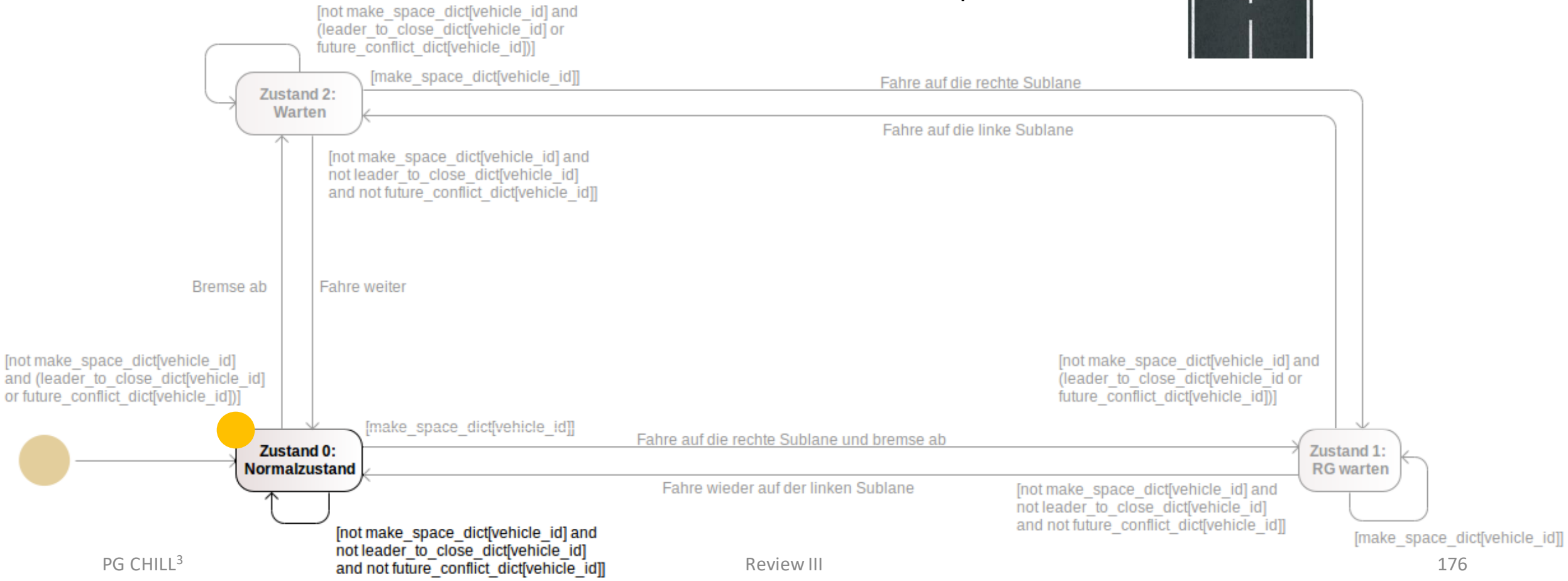




Zustandsautomat eines Fahrzeugs

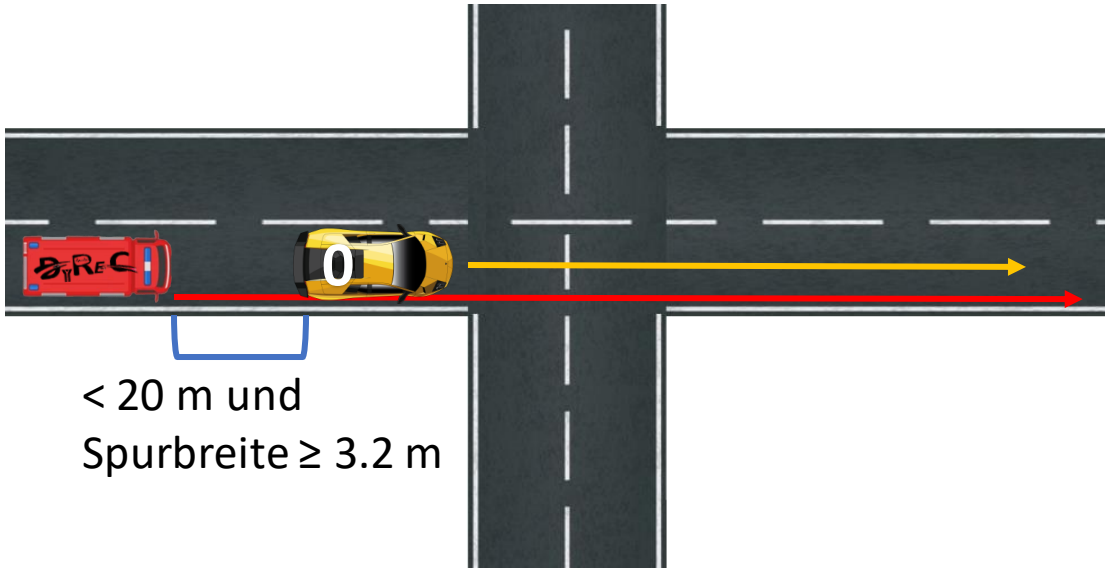
make_space_dict ={0: F}
leader_to_close_dict={0: F}
future_conflict_dict ={0: F}

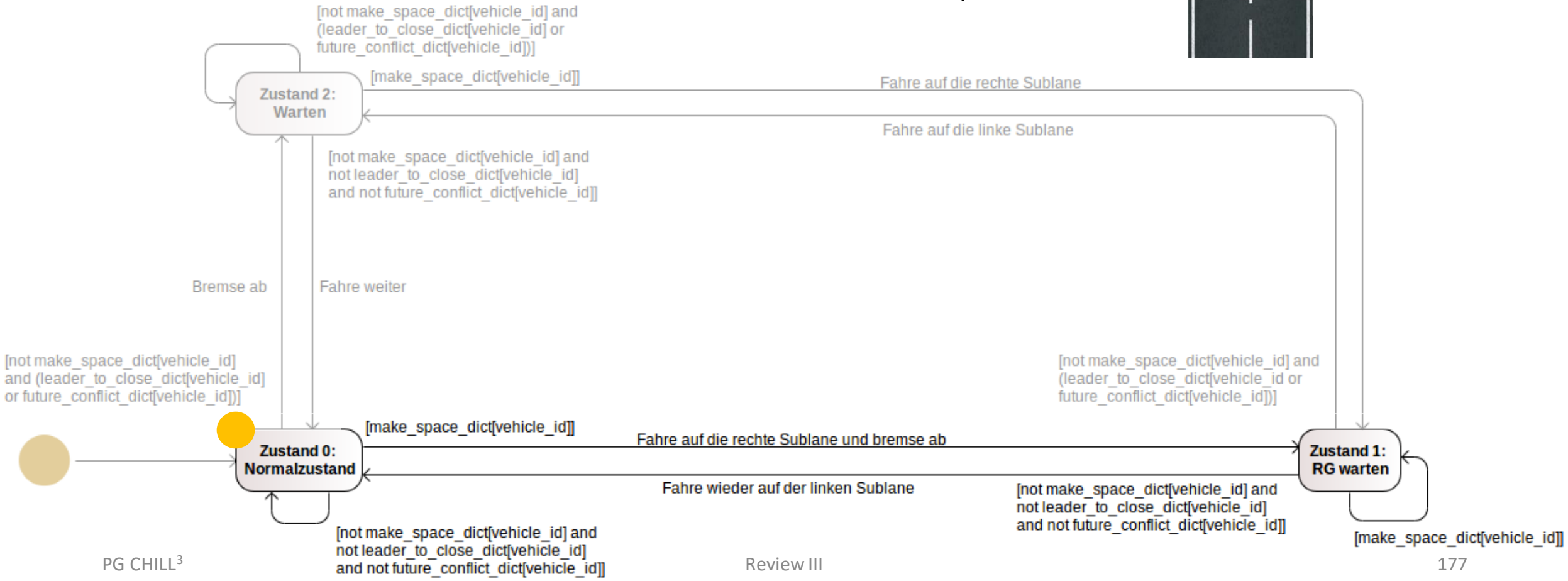




Zustandsautomat eines Fahrzeugs

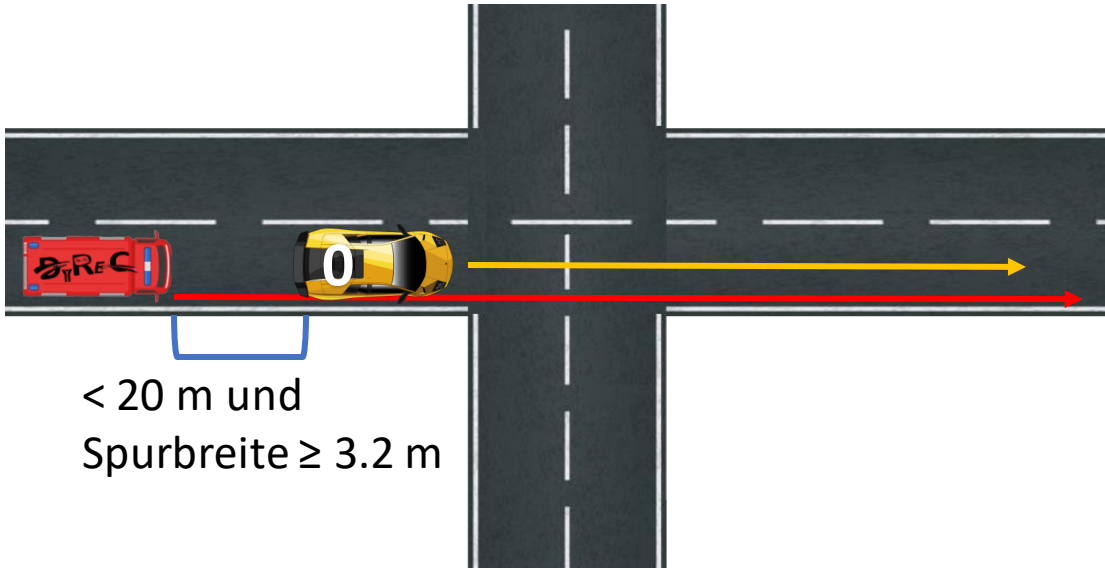
make_space_dict ={0: T}
leader_to_close_dict={0: F}
future_conflict_dict ={0: F}

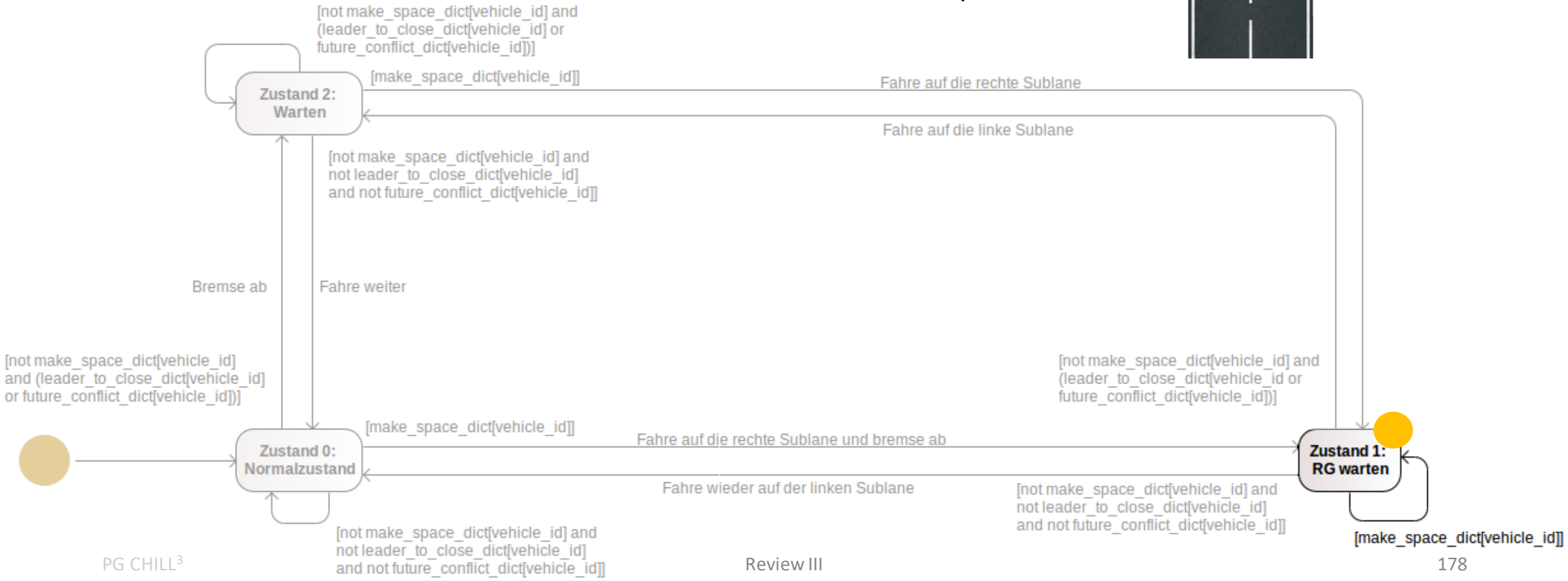




Zustandsautomat eines Fahrzeugs

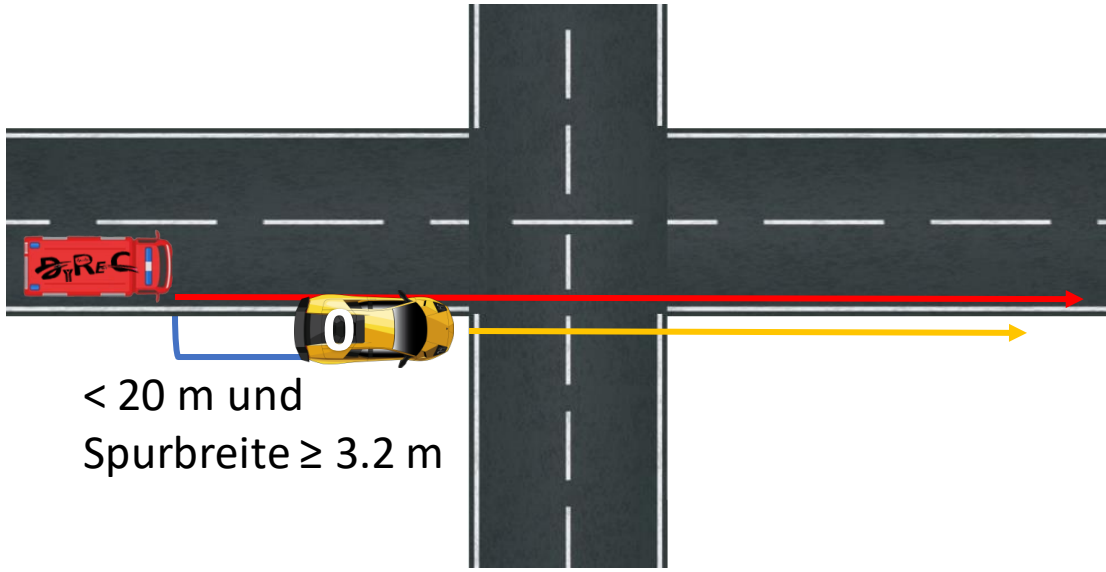
make_space_dict ={0: T}
leader_to_close_dict={0: F}
future_conflict_dict ={0: F}

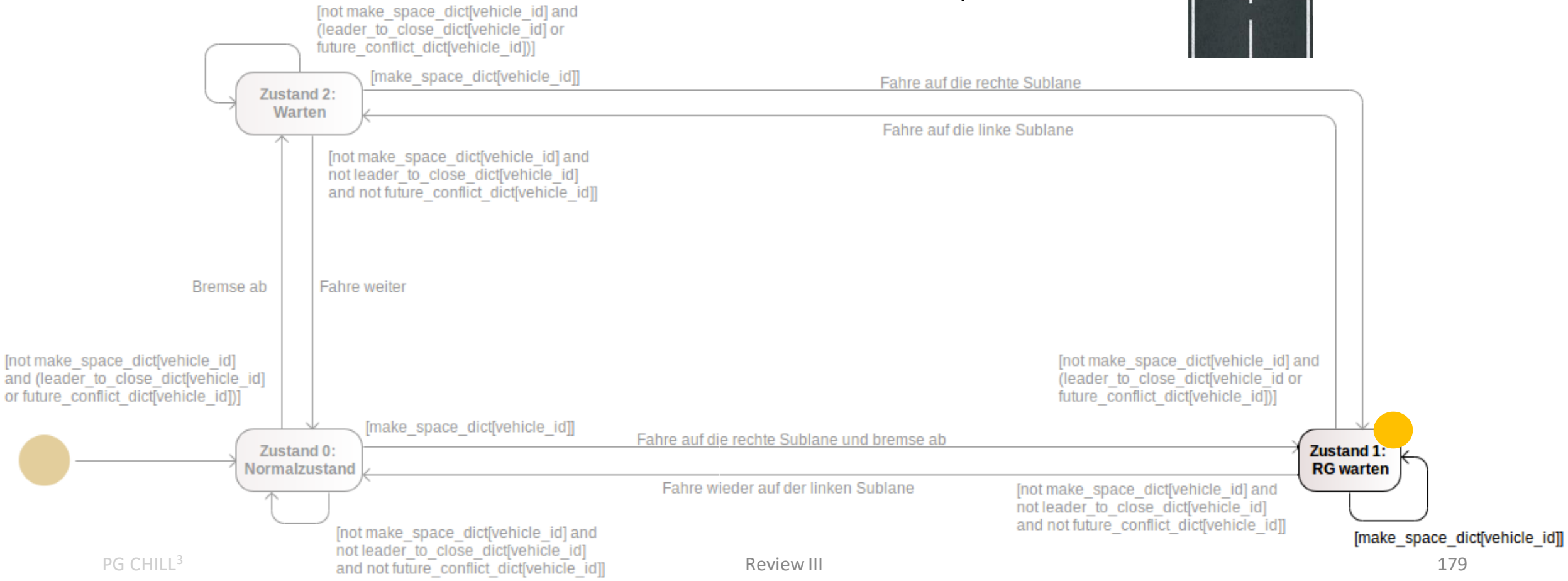




Zustandsautomat eines Fahrzeugs

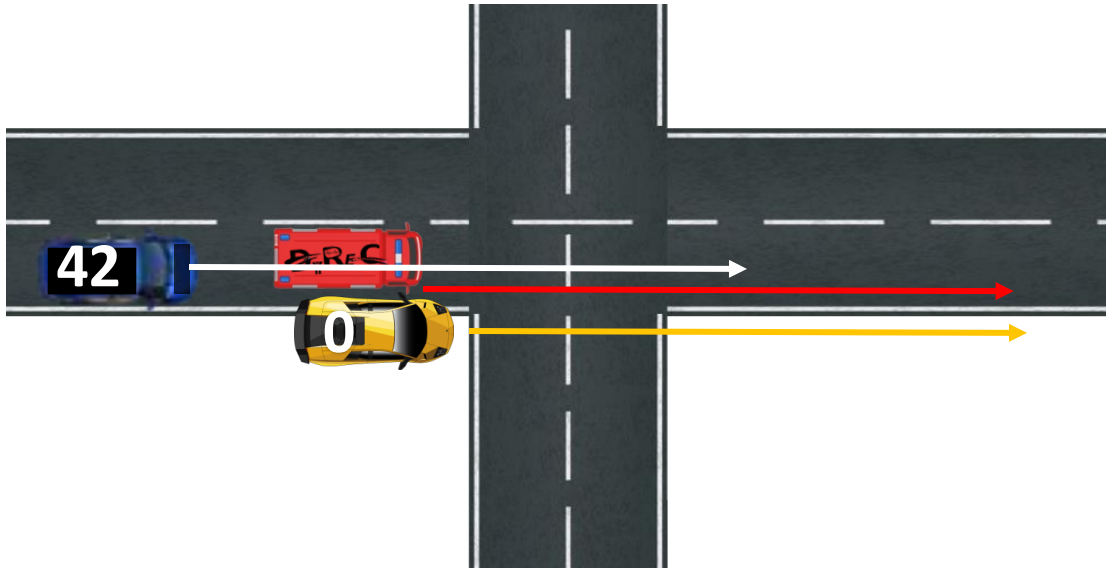
make_space_dict ={0: T}
leader_to_close_dict={0: F}
future_conflict_dict ={0: F}

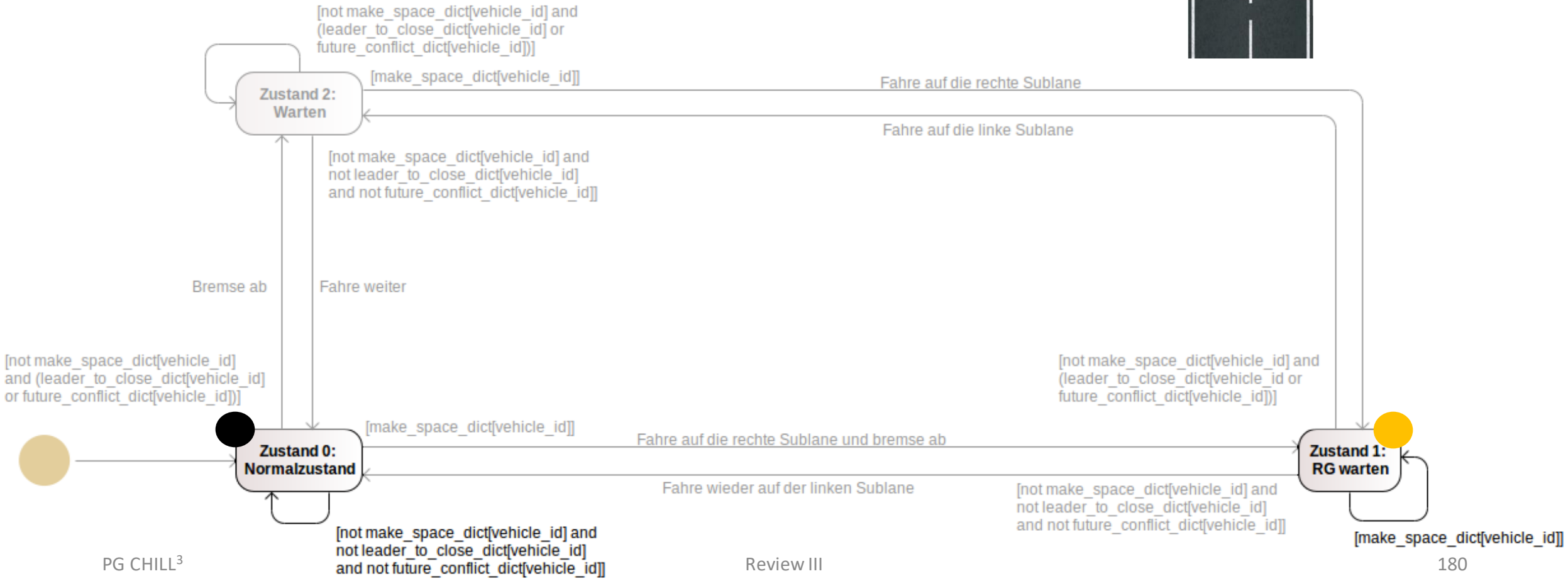




Zustandsautomat eines Fahrzeugs

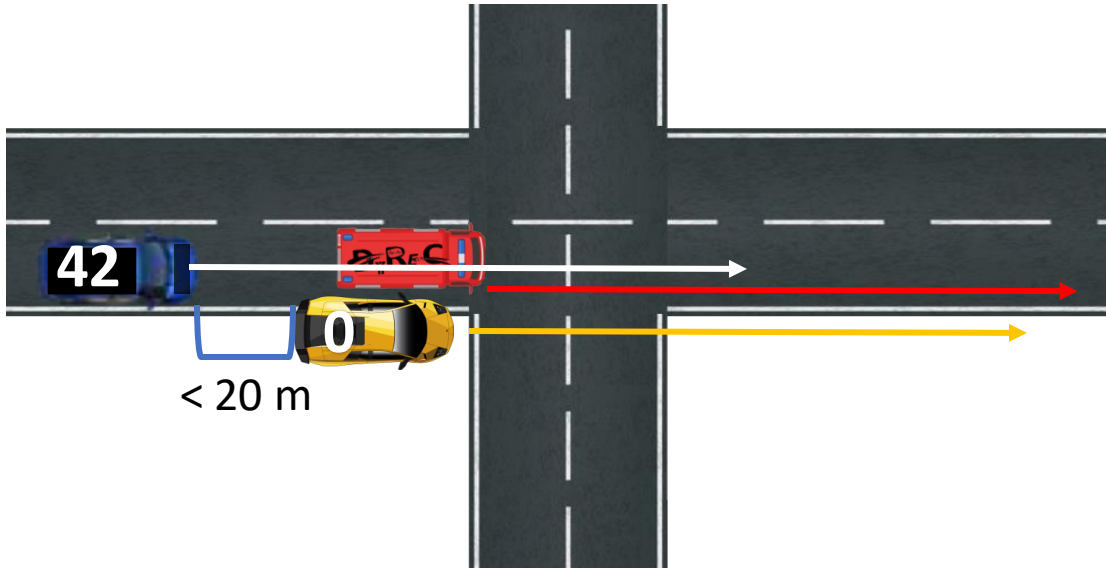
make_space_dict ={0: T, 42: F}
leader_to_close_dict={0: F, 42: F}
future_conflict_dict ={0: F, 42: F}





Zustandsautomat eines Fahrzeugs

make_space_dict ={0: T, 42: F}
leader_to_close_dict={0: F, 42: F}
future_conflict_dict ={0: F, 42: F}



[not make_space_dict[vehicle_id] and (leader_to_close_dict[vehicle_id] or future_conflict_dict[vehicle_id])]

[make_space_dict[vehicle_id]]

Zustand 2: Warten

Fahre auf die rechte Sublane

Fahre auf die linke Sublane

[not make_space_dict[vehicle_id] and not leader_to_close_dict[vehicle_id] and not future_conflict_dict[vehicle_id]]

Bremse ab

Fahre weiter

[not make_space_dict[vehicle_id] and (leader_to_close_dict[vehicle_id] or future_conflict_dict[vehicle_id])]

[not make_space_dict[vehicle_id] and (leader_to_close_dict[vehicle_id or future_conflict_dict[vehicle_id])]

[make_space_dict[vehicle_id]]

Zustand 0: Normalzustand

Fahre auf die rechte Sublane und bremse ab

Zustand 1: RG warten

Fahre wieder auf der linken Sublane

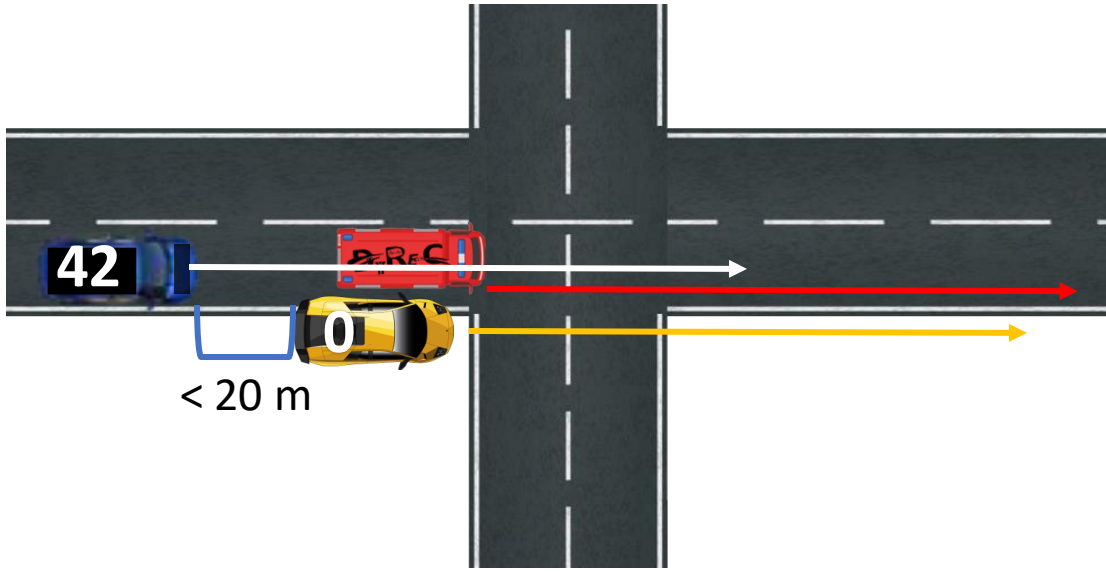
[not make_space_dict[vehicle_id] and not leader_to_close_dict[vehicle_id] and not future_conflict_dict[vehicle_id]]

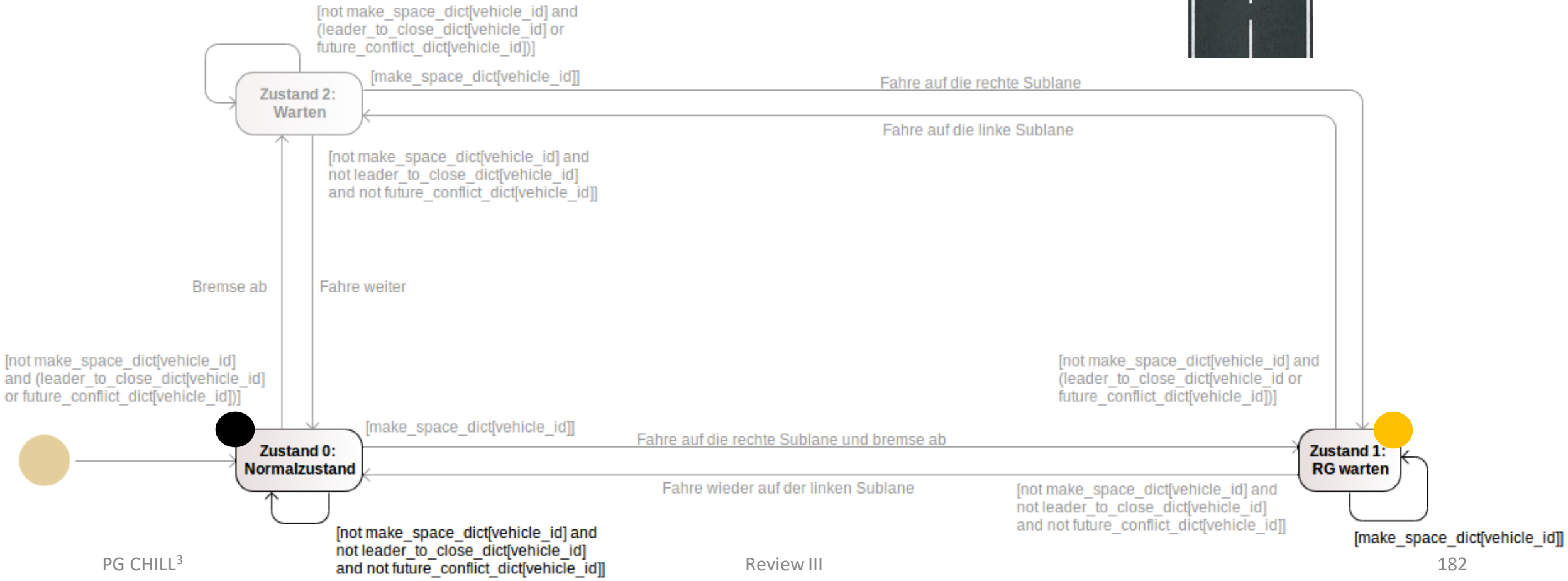
[not make_space_dict[vehicle_id] and not leader_to_close_dict[vehicle_id] and not future_conflict_dict[vehicle_id]]

[make_space_dict[vehicle_id]]

Zustandsautomat eines Fahrzeugs

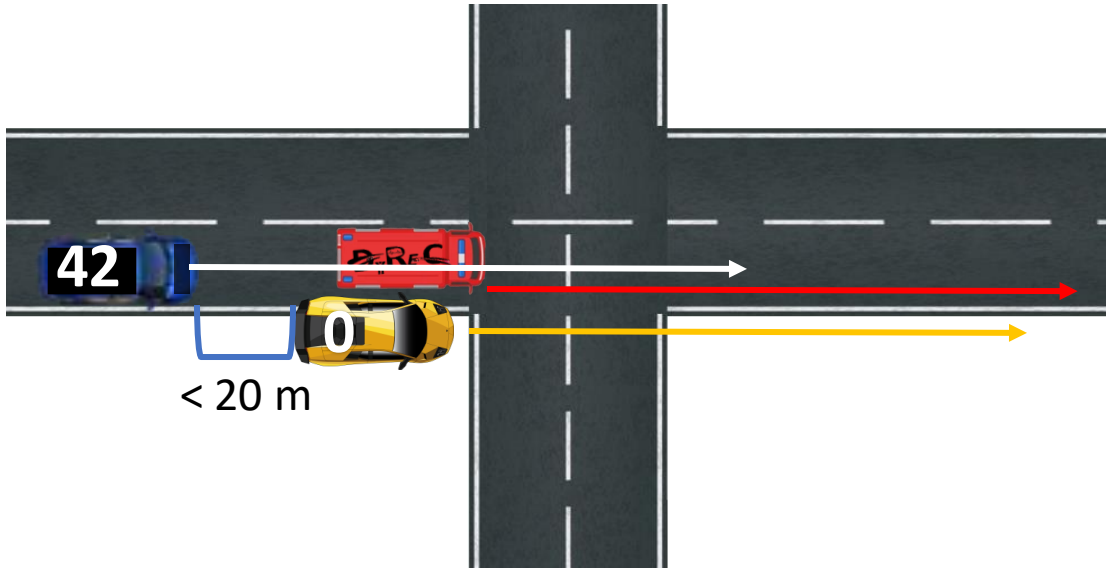
make_space_dict ={0: T, 42: F}
leader_to_close_dict={0: F, 42: T}
future_conflict_dict ={0: F, 42: F}

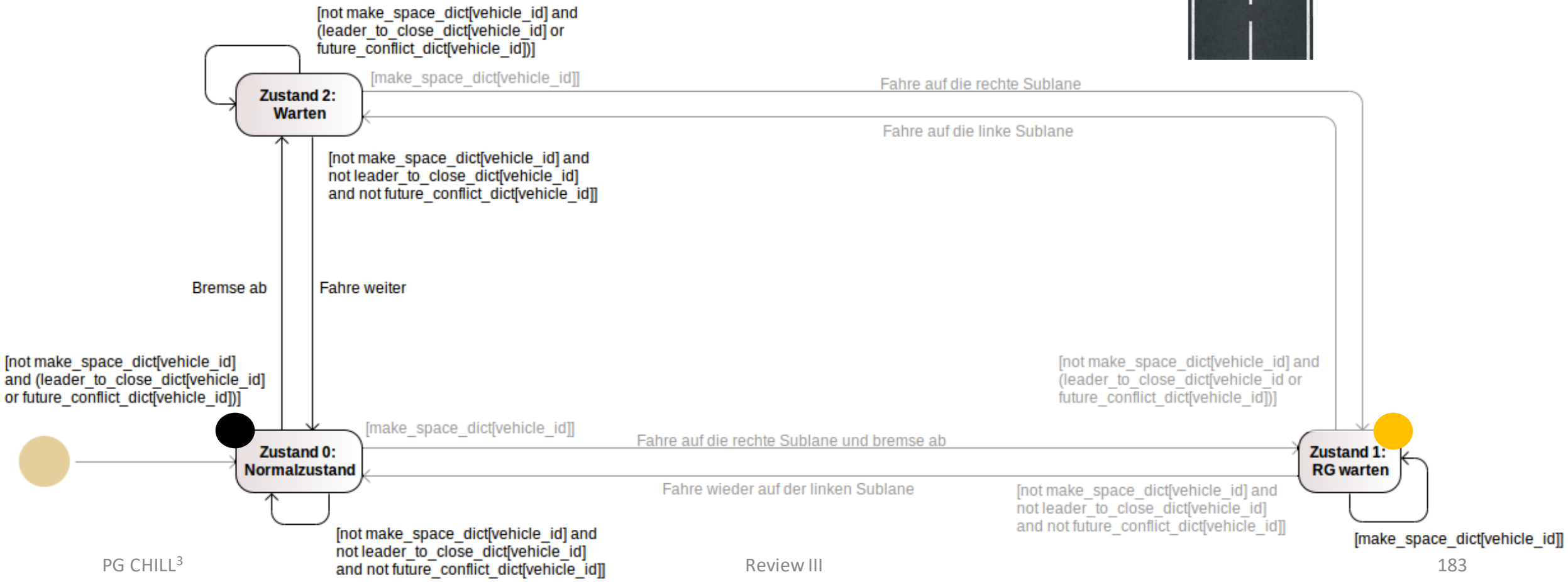




Zustandsautomat eines Fahrzeugs

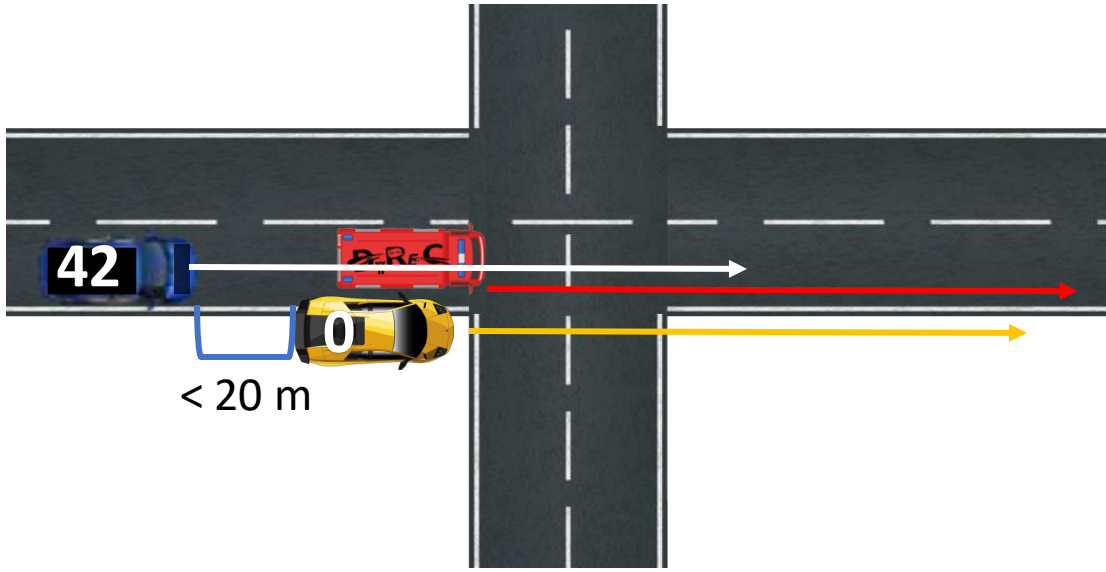
make_space_dict ={0: T, 42: F}
leader_to_close_dict={0: F, 42: T}
future_conflict_dict ={0: F, 42: F}

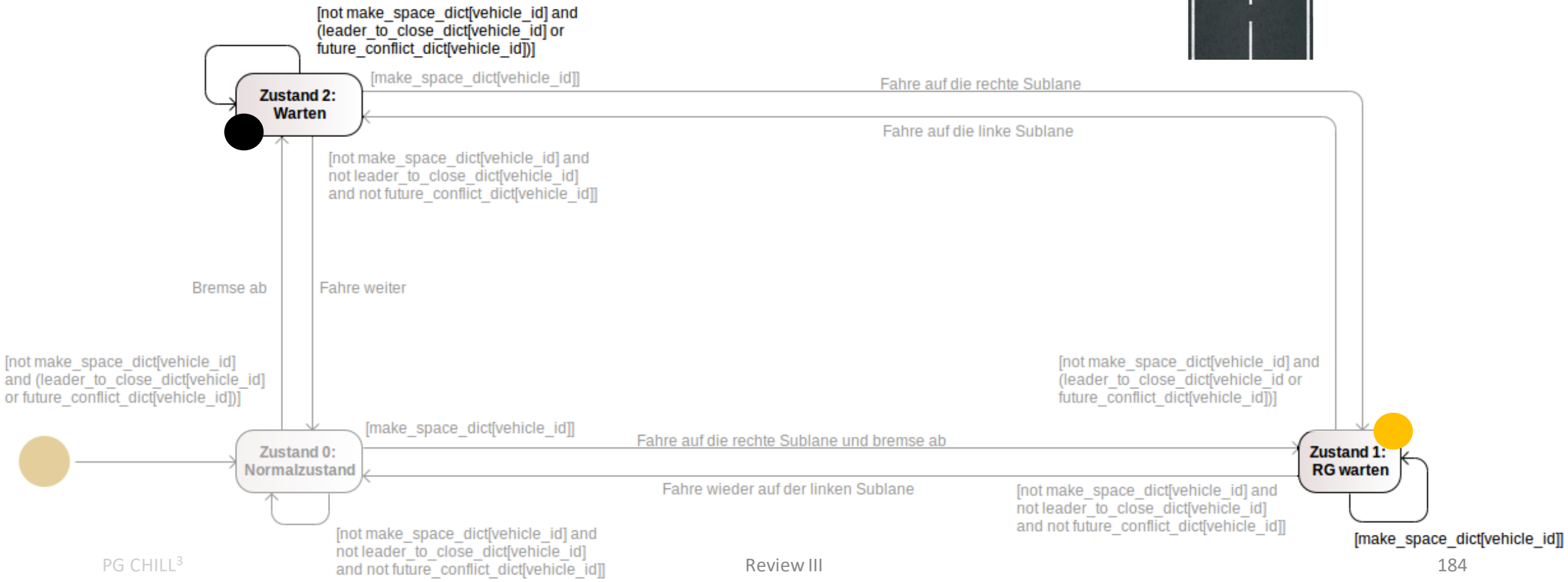




Zustandsautomat eines Fahrzeugs

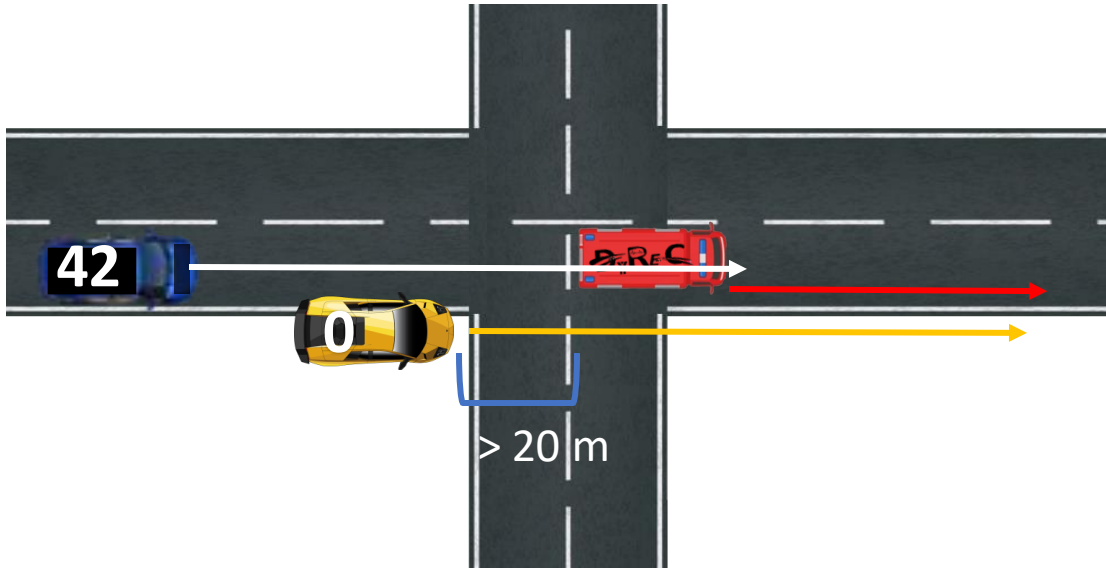
make_space_dict ={0: T, 42: F}
leader_to_close_dict={0: F, 42: T}
future_conflict_dict ={0: F, 42: F}

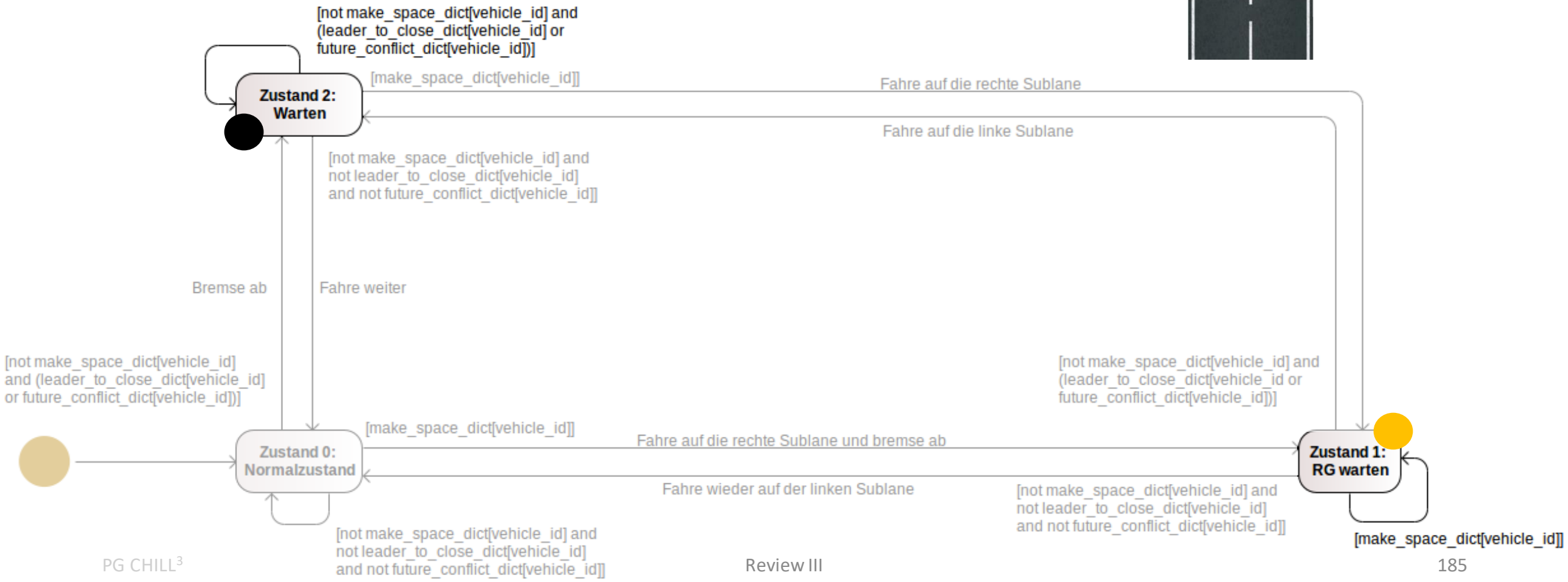




Zustandsautomat eines Fahrzeugs

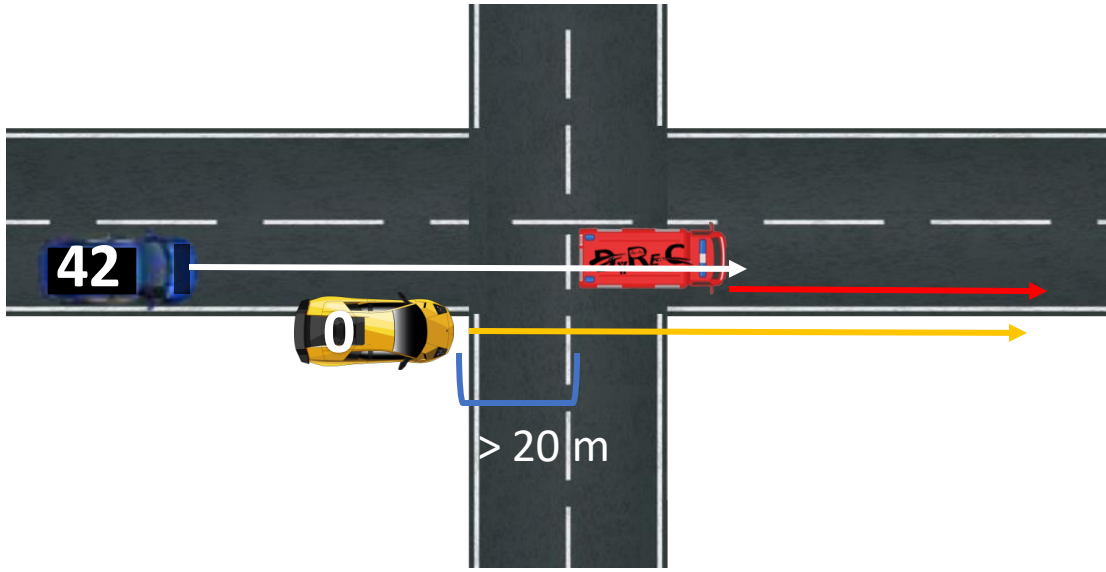
make_space_dict ={0: T, 42: F}
leader_to_close_dict={0: F, 42: T}
future_conflict_dict ={0: F, 42: F}

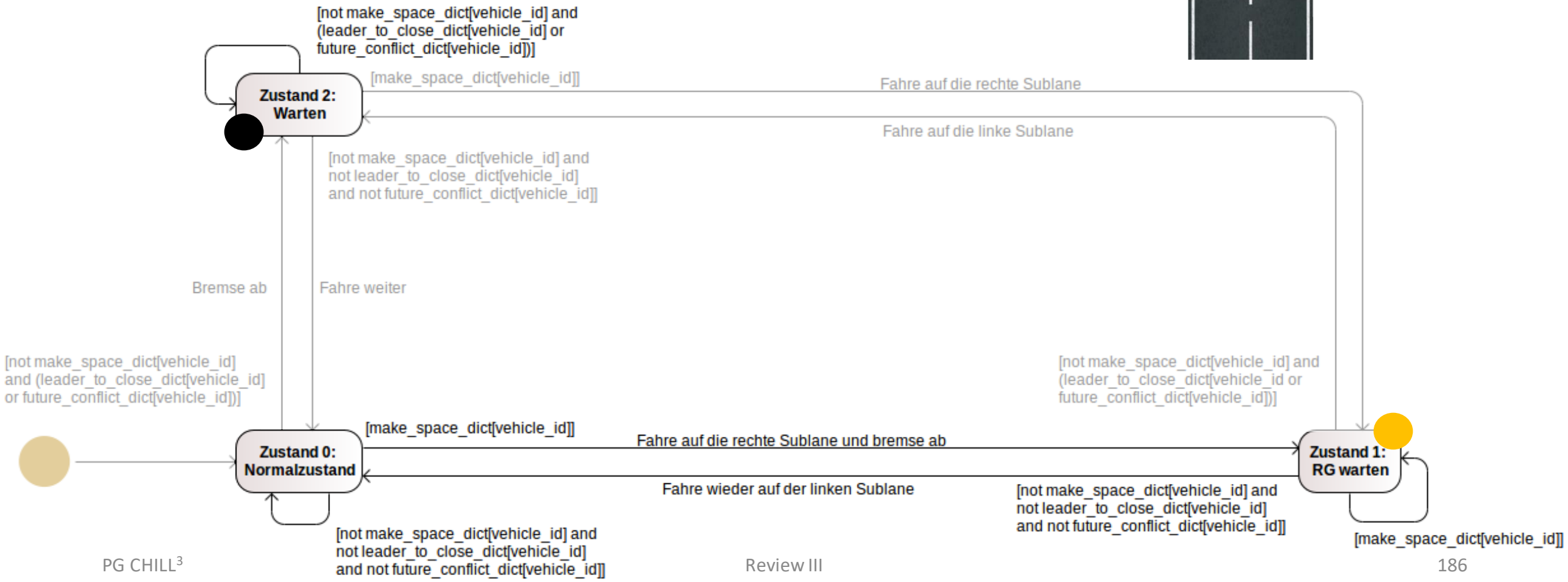




Zustandsautomat eines Fahrzeugs

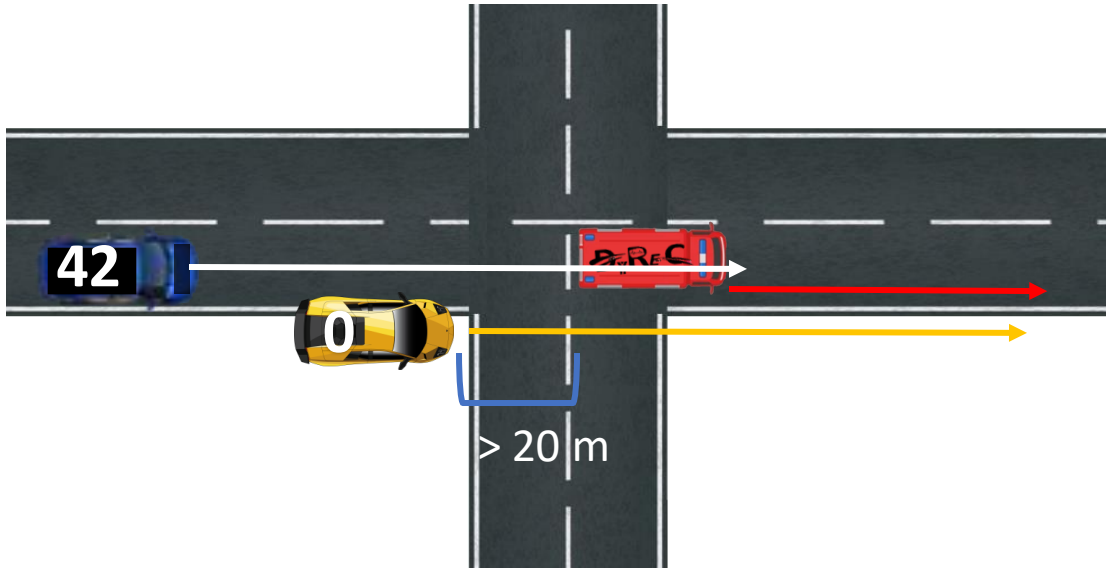
make_space_dict ={0: F, 42: F}
leader_to_close_dict={0: F, 42: T}
future_conflict_dict ={0: F, 42: F}

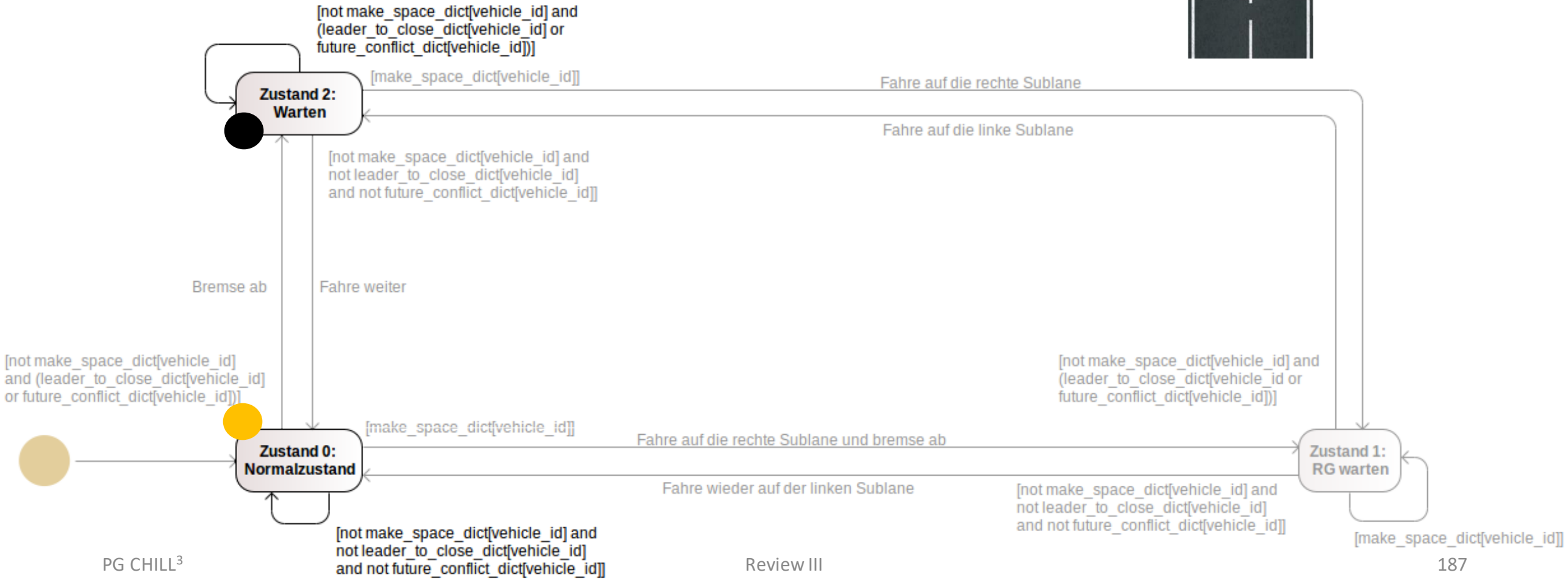




Zustandsautomat eines Fahrzeugs

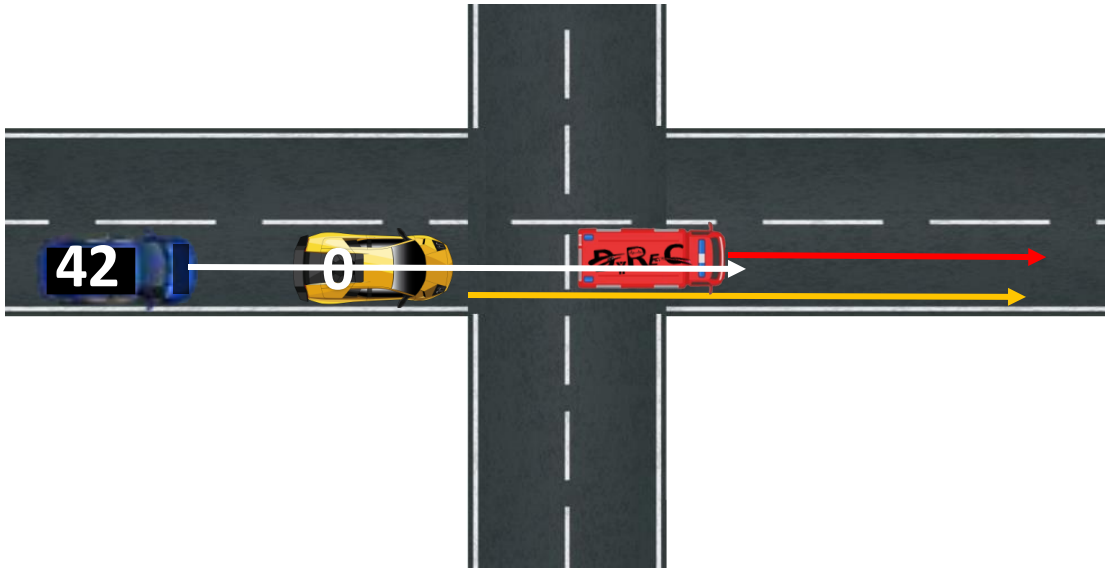
make_space_dict ={0: F, 42: F}
leader_to_close_dict={0: F, 42: T}
future_conflict_dict ={0: F, 42: F}

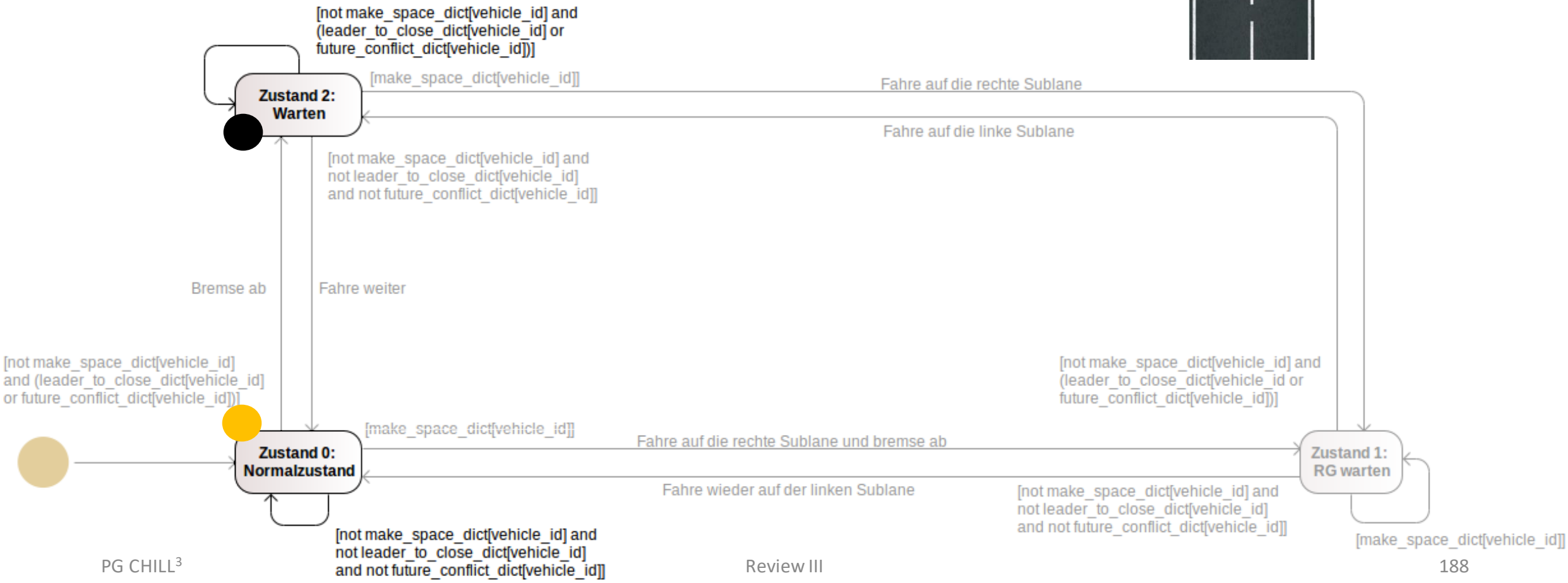




Zustandsautomat eines Fahrzeugs

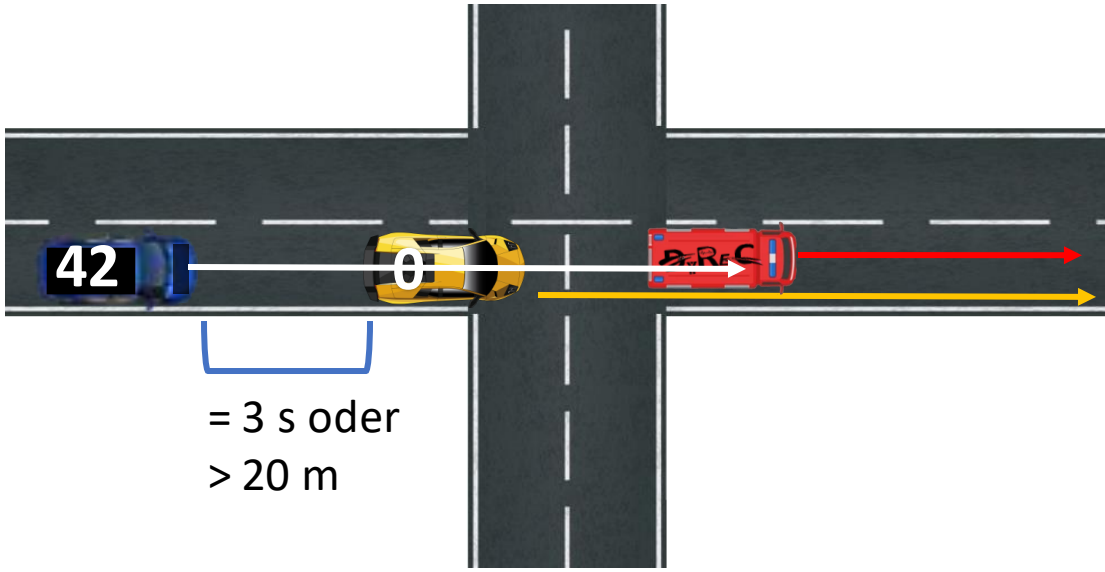
make_space_dict ={0: F, 42: F}
leader_to_close_dict={0: F, 42: T}
future_conflict_dict ={0: F, 42: F}

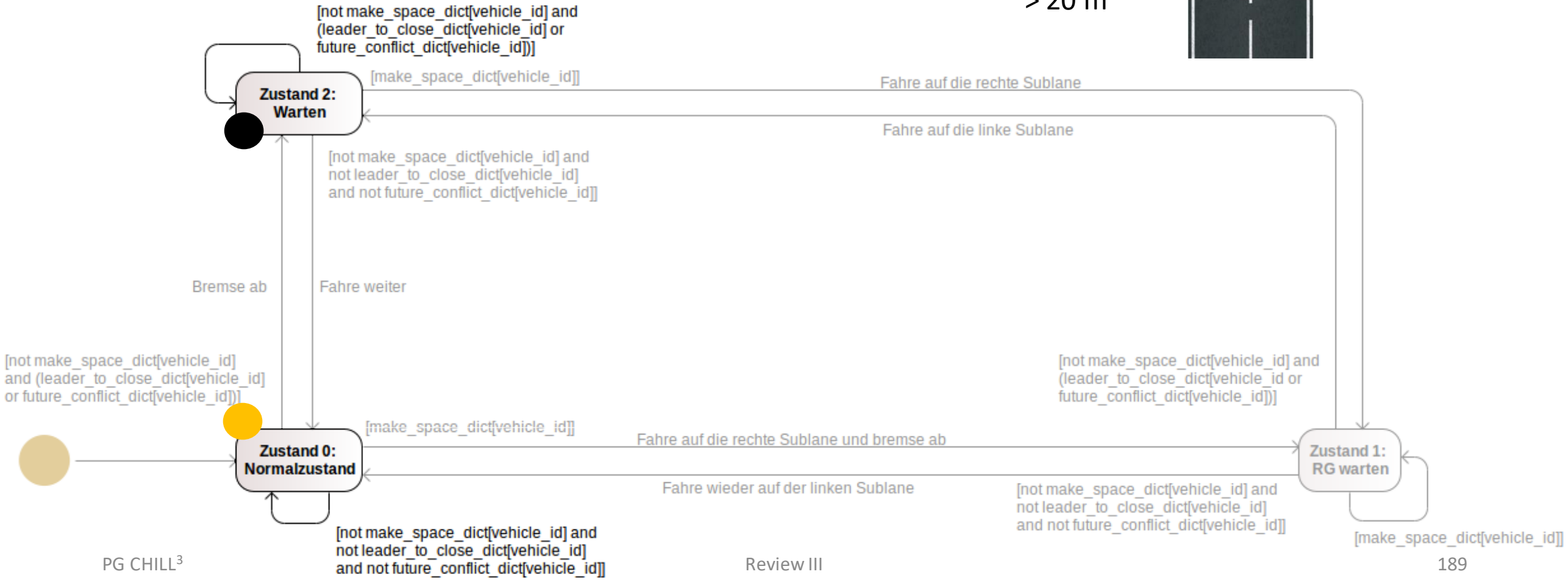




Zustandsautomat eines Fahrzeugs

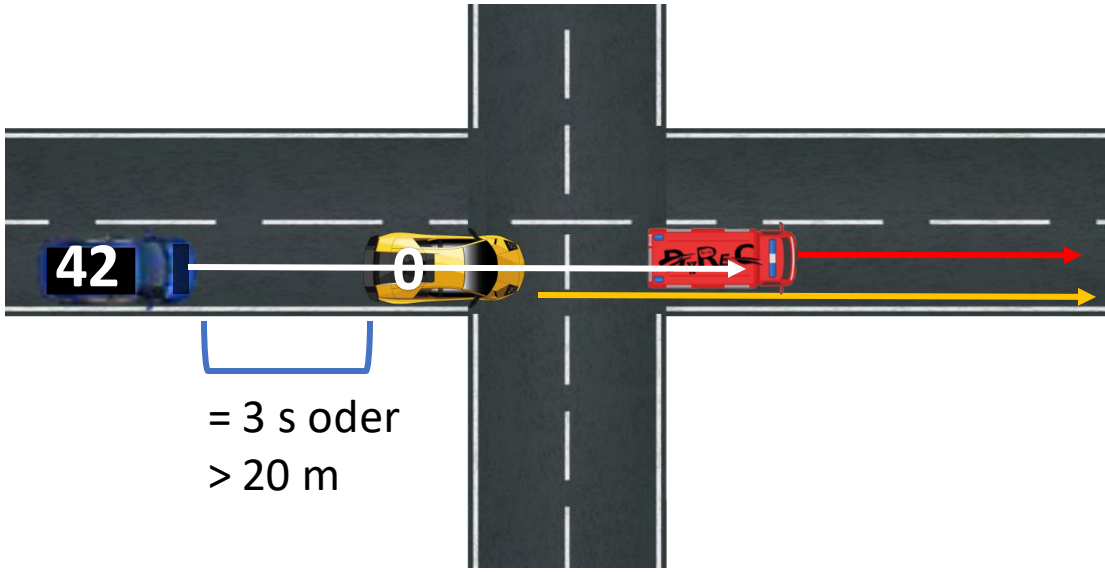
make_space_dict ={0: F, 42: F}
leader_to_close_dict={0: F, 42: T}
future_conflict_dict ={0: F, 42: F}

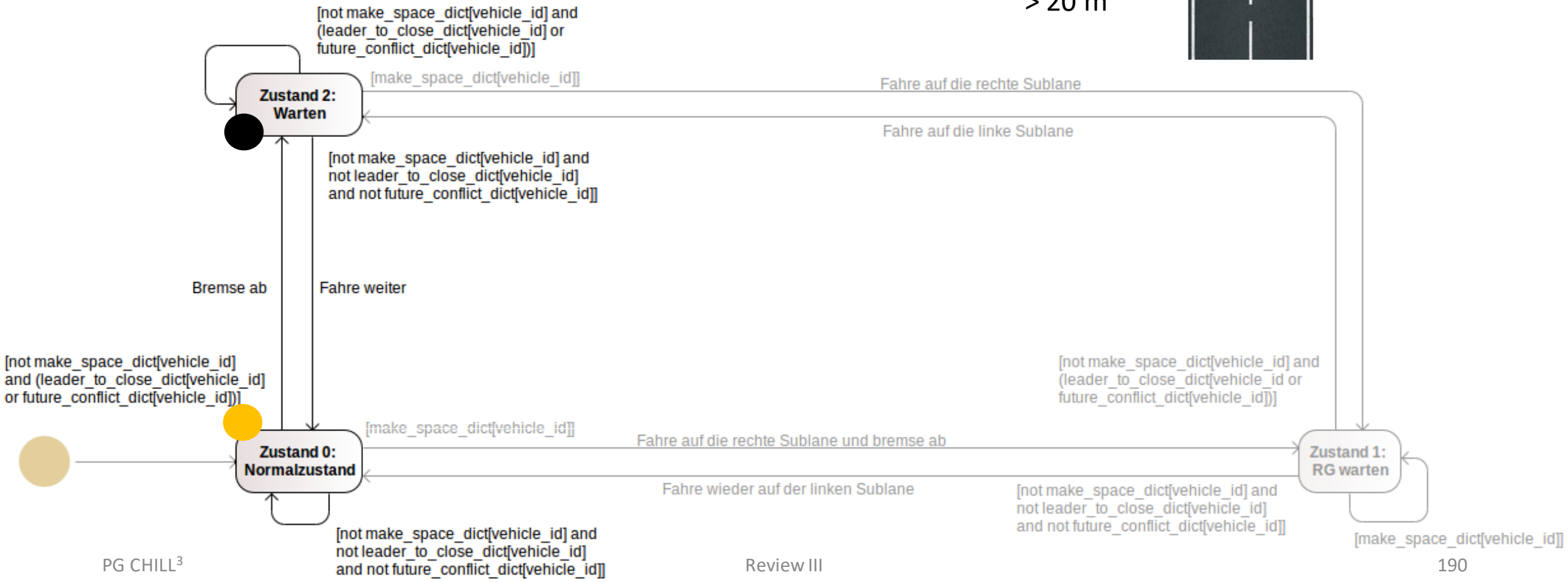




Zustandsautomat eines Fahrzeugs

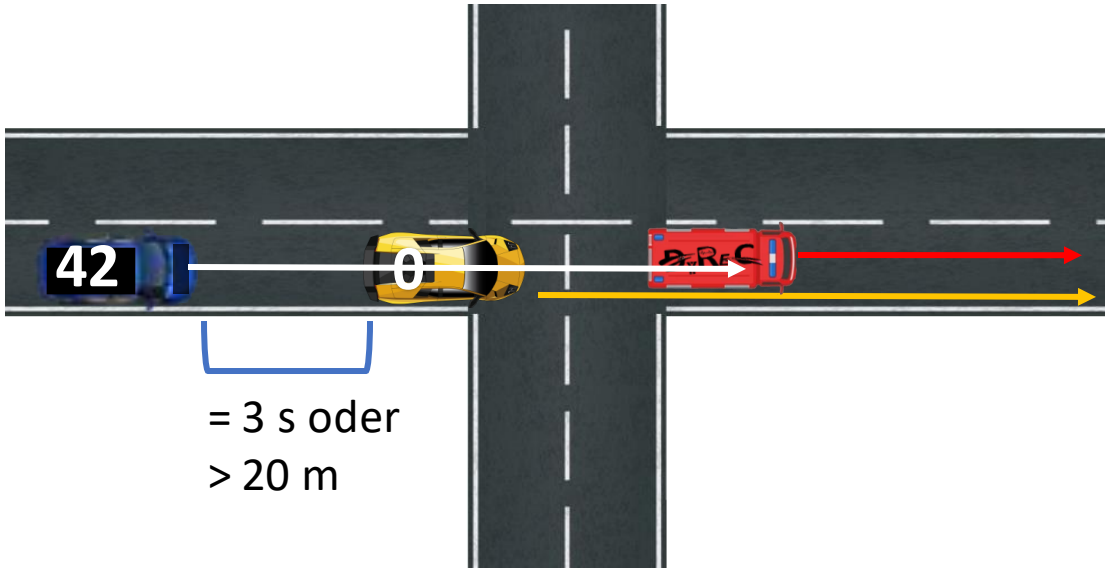
make_space_dict ={0: F, 42: F}
leader_to_close_dict={0: F, 42: F}
future_conflict_dict ={0: F, 42: F}

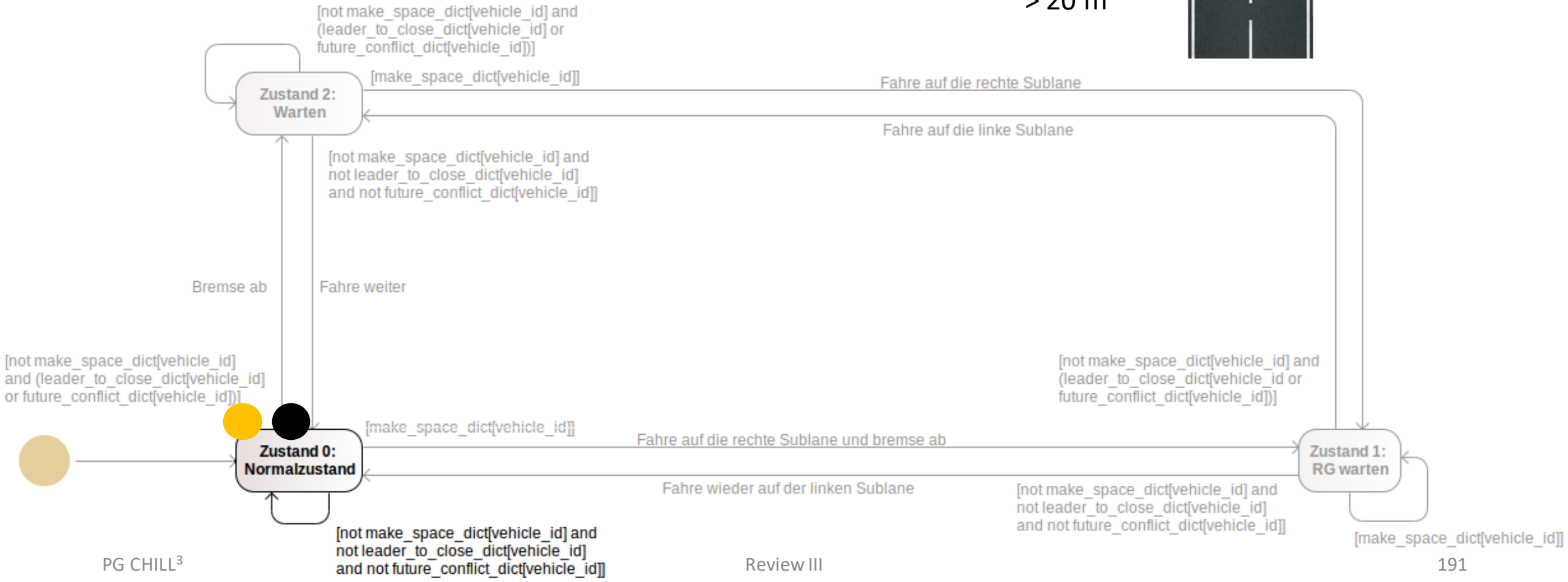




Zustandsautomat eines Fahrzeugs

make_space_dict ={0: F, 42: F}
leader_to_close_dict={0: F, 42: F}
future_conflict_dict ={0: F, 42: F}





Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
  - SUMO-Umgebung
  - Komponentendiagramm
  - Programmablauf
  - Kaskadierendes Verfahren
  - LSA-Anpassung
  - Dynamische Rettungsgasse
  - Tests
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Unittests

Referenzsystem

Klasse	Coverage
ELR	97,8 %
Controller	52,6 %
Emergency_Vehicles	100,0 %
SUMO_abstraction	100,0 %
Total	**88,2 %**

Unittests

Referenzsystem

Klasse	Coverage
ELR	97,8 %
Controller	52,6 %
Emergency_Vehicles	100,0 %
SUMO_abstraction	100,0 %
Total	**88,2 %**

DyReC

Klasse	Coverage
ELR	97,8 %
Controller	15,6 %
Emergency_Vehicles	100,0 %
SUMO_abstraction	85,0 %
DLVMS	26,3 %
DRW	0,0 %
ID_Part	0,0 %
Exceptions	87,5 %
Total	**40,3 %**

Integrationstests

Referenzsystem

Klasse	Coverage
ELR	91,3 %
Controller	94,9 %
Emergency_Vehicles	64,0 %
SUMO_abstraction	89,4 %
Exceptions	66,7 %
Total	**89,8 %**

Integrationstests

Referenzsystem

Klasse	Coverage
ELR	91,3 %
Controller	94,9 %
Emergency_Vehicles	64,0 %
SUMO_abstraction	89,4 %
Exceptions	66,7 %
Total	**89,8 %**

DyReC

Klasse	Coverage
ELR	91,3 %
Controller	97,2 %
Emergency_Vehicles	64,0 %
SUMO_abstraction	90,7 %
DLVMS	70,8 %
DRW	100,0 %
ID_Part	100,0 %
Exceptions	83,3 %
Total	**84,0 %**

Systemtests

- 3 Systemtests, zur Prüfung der Funktionalität des Systems
- Beispielrettungsgasse & kaskadierendes Verfahren
- Rettungsgasse kann nicht gebildet werden
- LSA-Anpassung => fehlgeschlagen
- => zu geringe Fallabdeckung

GUI Tests

- 7 GUI Tests
- Beschreibung Workflow zur Nutzung der GUI
- Fehlerfälle
- => gute Abdeckung

Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Übersicht umgesetzte Features

- ELR zur Simulation verschiedener Notfallszenarien und Orte
  - Funktioniert mit jeder SUMO Karte, solange Informationen zu Notfällen, etc. bereitgestellt werden
- Referenzsystem* von SUMO
  - Funktioniert mit jeder SUMO Karte
- DyReC-Implementierung des kaskadierenden Verfahrens sowie der dynamischen Rettungsgasse
  - Funktioniert nur mit "einfachen" Karten
  - Große Probleme mit Export aus OSM => funktioniert nicht mit Basiskarte

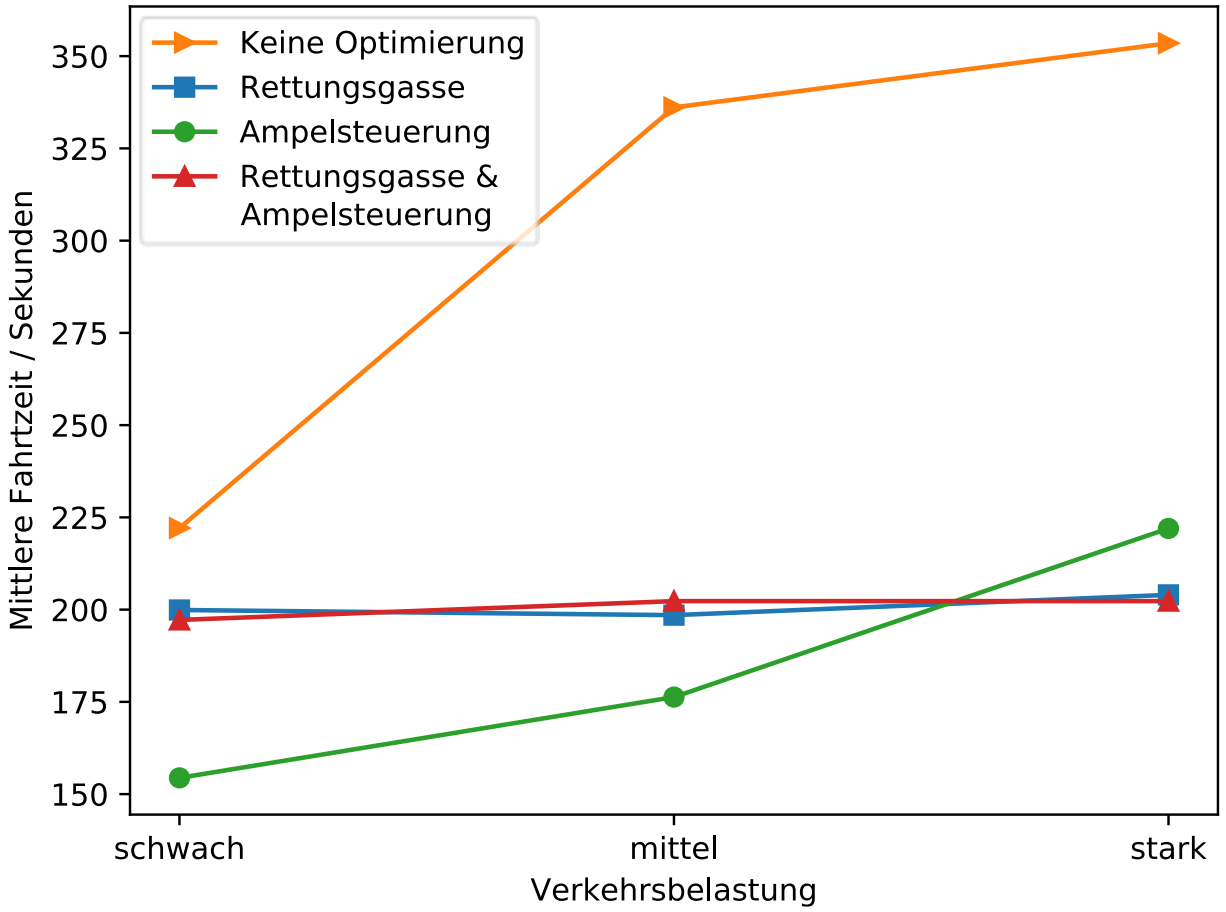
[* https://sumo.dlr.de/docs/Simulation/Emergency.html (13. 08. 2020)]

Demonstrator

Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Vorstudie

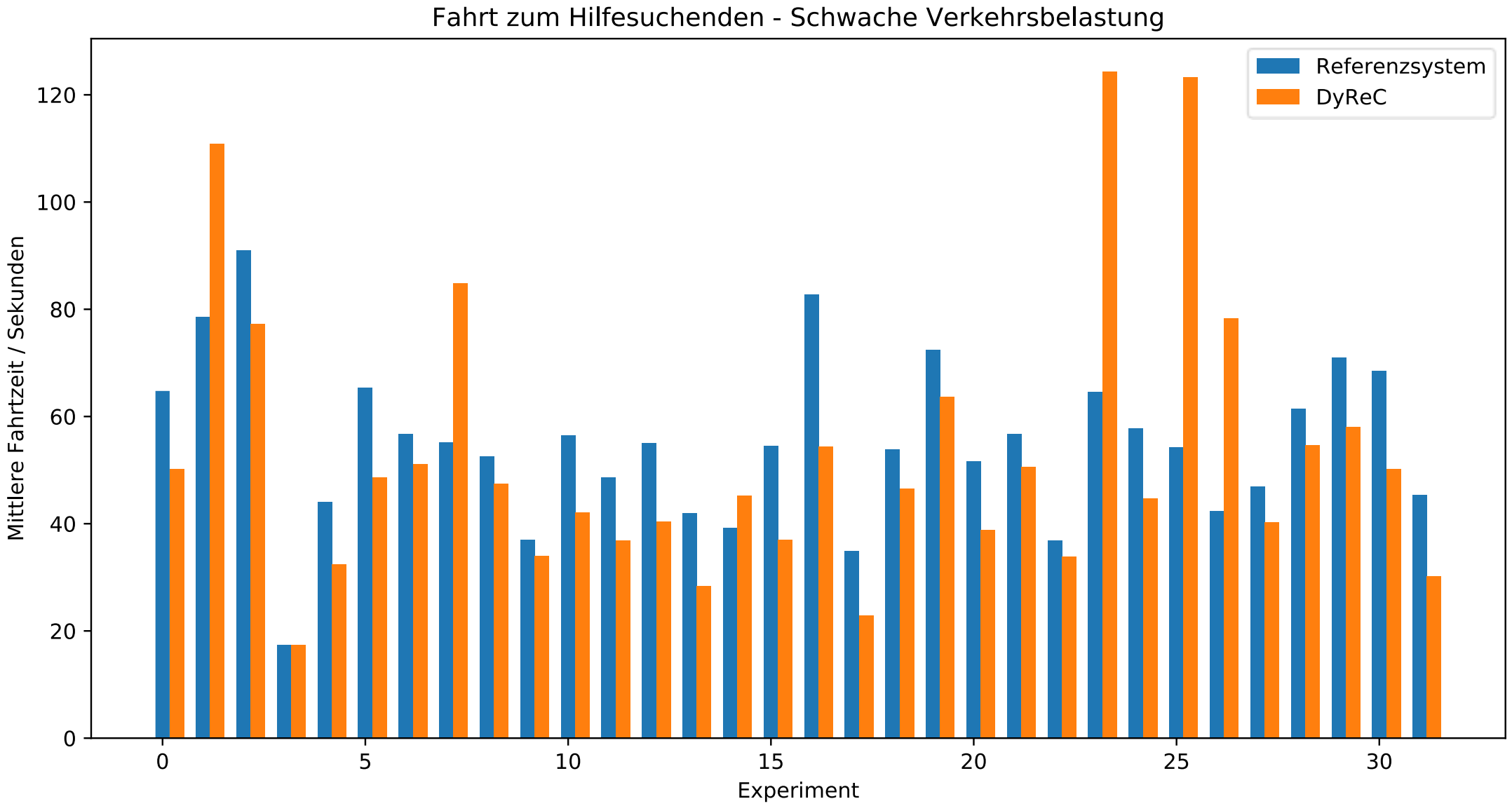


- Klassische Rettungsgasse
- Instantane Ampelsteuerung

Ergebnisse schwache Verkehrsbelastung

- 32 Experimente (Ziele / Notfälle)
- 3 Verschiedene Verkehrsbelastungen
- 10 Verschiedene Seeds zur Generierung zufälliger Szenarien
- Exemplarische Bestimmung der Equilibrierungsdauer

Ergebnisse schwache Verkehrsbelastung



Mögliche Gründe für die schlechte Performanz von DyReC

- Keine Zeit für Tests der Heuristiken der dynamischen Rettungsgasse
- => Schlechte Heuristiken?
- Zu geringe Abdeckung von Cornercases
- => Deadlocks?

Wie hätte man es besser machen können?

- Keine initiale Nutzung einer Karte mit OSM-Export
- Effizienter und schneller programmieren
- => mehr Zeit für Tests und Entwicklung von Heuristiken

Agenda

- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Demonstrator
- Evaluation
- Ausblick
- Fazit

Ausblick

- Verbesserung der Logik zur Bildung der dynamischen Rettungsgasse
  - Ggf. mittels restringierter Optimierungsstrategie
- Einbezug von mehr Informationen in die Bildung der Rettungsgasse
  - Verzahnung von makroskopischer – und mikroskopischer Verkehrssicht
  - Weiterentwicklung des KI-Konzepts zur Verbesserung der Vorhersage des Verkehrsflusses
- Stärkerer Einbezug der Verkehrsteilnehmer bei der Bildung der dynamischen Rettungsgasse

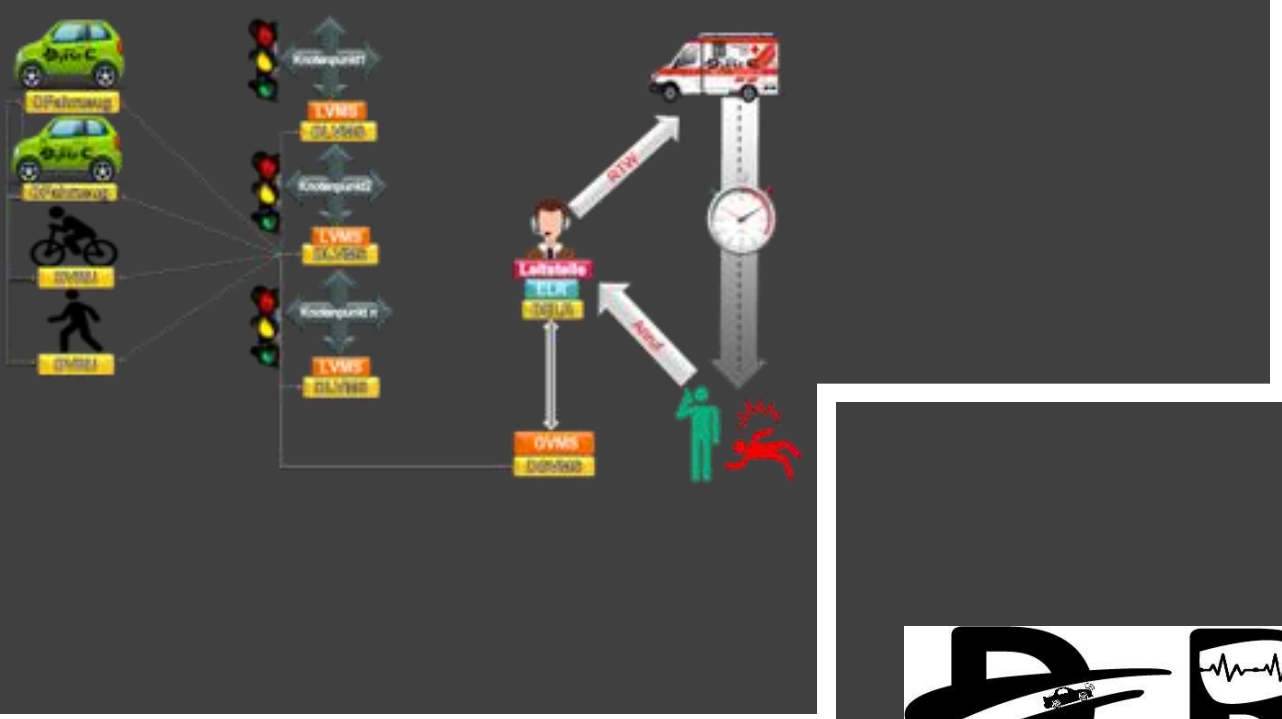
Agenda

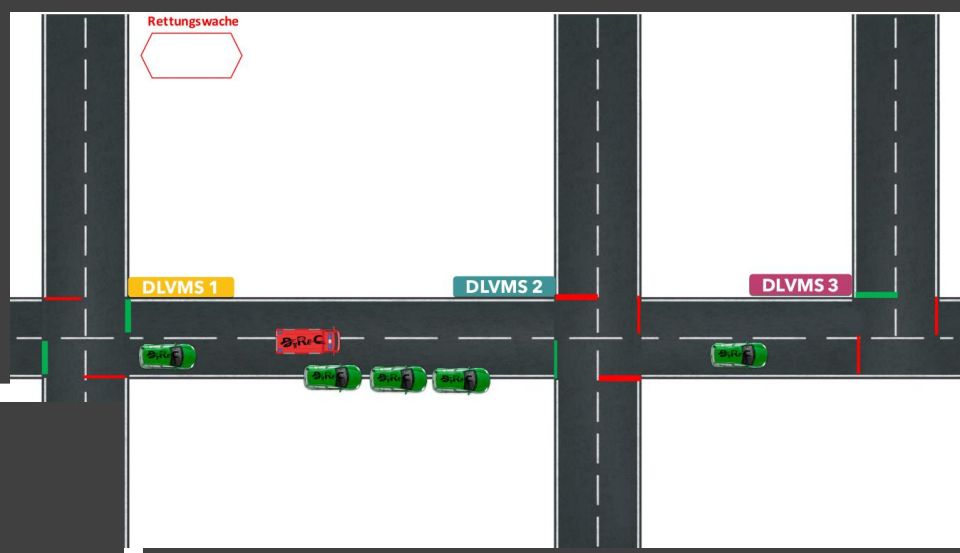
- Vision
- Projektmanagement
- Initialer Systemaufbau
- Use Cases
- Konzepte
- Implementierung
- Demonstrator
- Evaluation
- Ausblick
- Fazit

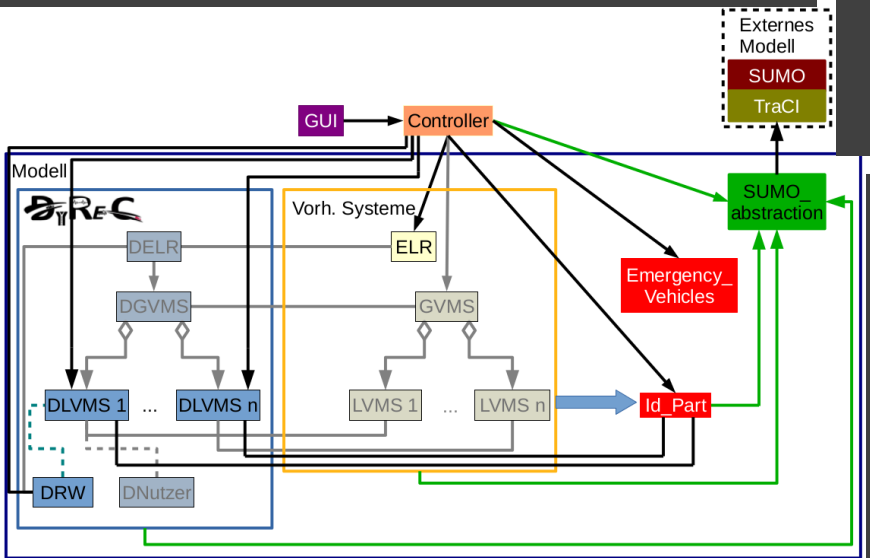
Fazit

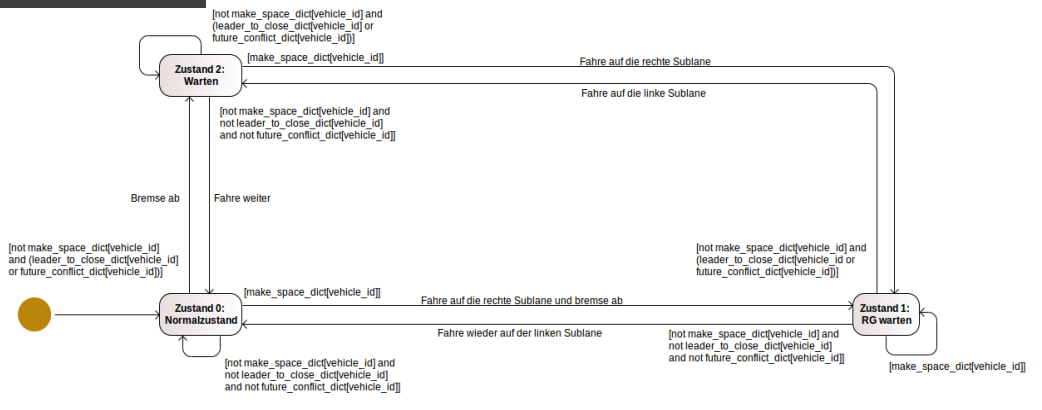
- Projektidee
- Erweiterbare Architektur
- Abbildung der realen Alarmierung
- Erweiterbare Austauschformate
- Aktuelle Corona-Umstände
- Jira Nutzung zum Ende besser

- Fachliche Probleme
- Single Point of Failure
- Zeitplanung & Termineinhaltung
- Scrum -> geringe Nutzung
- Dokumentation von Aufgaben
- Transparenz
- Geringe Gruppengröße


LVMS
LVMS
LVMS
GVMS
RTW
Leitstelle
ELR
Hund

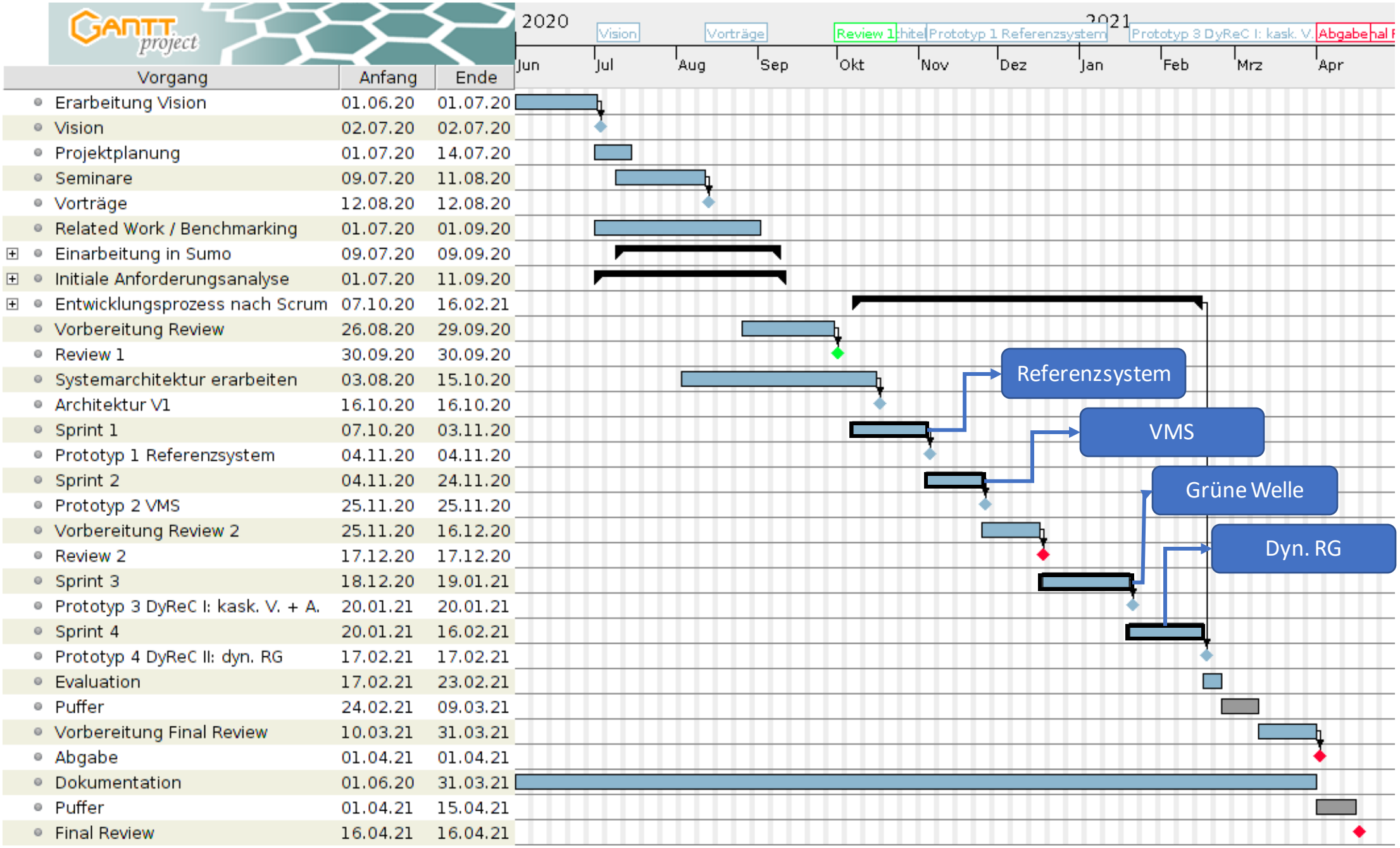

Rettungswache
DLVMS 1
DLVMS 2
DLVMS 3


Externes Modell
SUMO
TraCI
GUI
Controller
Modell
Vorh. Systeme
SUMO_ abstraction
Emergency_ Vehicles
DELR
ELR
DGVMS
GVMS
DLVMS 1
DLVMS n
LVMS 1
LVMS n
Id_Part
DRW
DNutzer

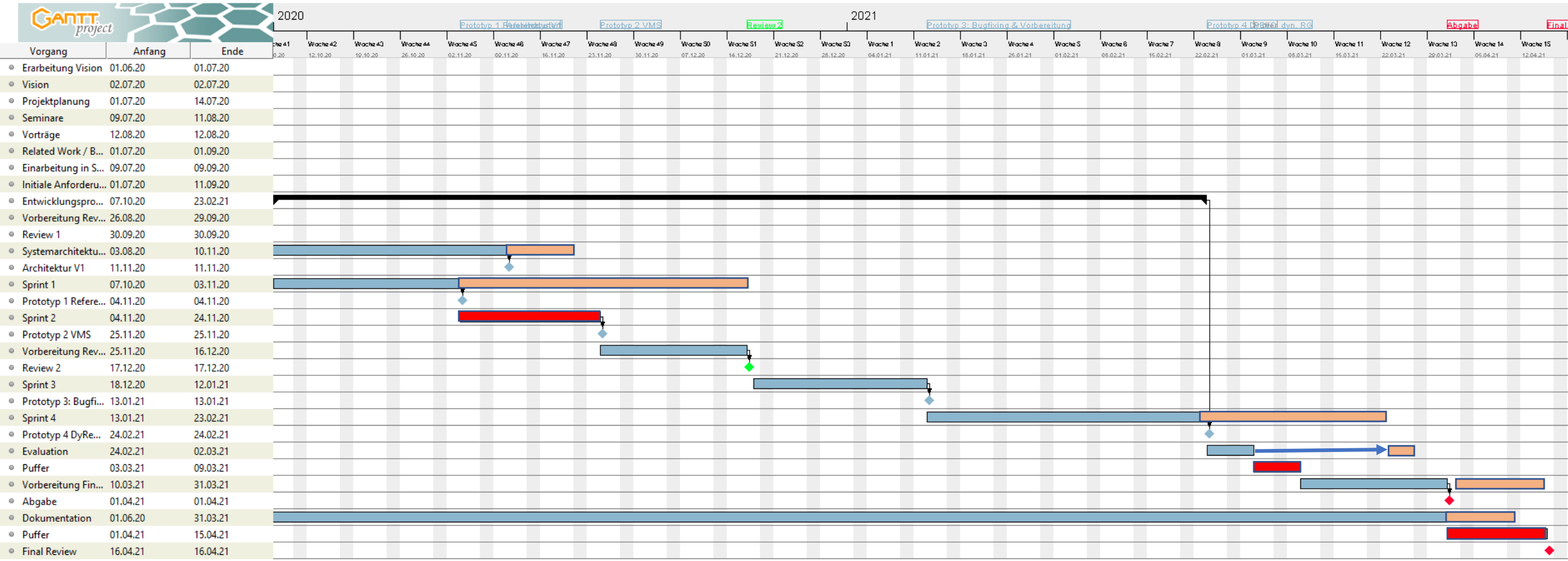

[not make_space_dict[vehicle_id] and (leader_to_close_dict[vehicle_id] or future_conflict_dict[vehicle_id])]
Zustand 2: Warten
[make_space_dict[vehicle_id]]
Fahre auf die rechte Sublane
Fahre auf die linke Sublane
[not make_space_dict[vehicle_id] and not leader_to_close_dict[vehicle_id] and not future_conflict_dict[vehicle_id]]
Bremse ab
Fahre weiter
[not make_space_dict[vehicle_id] and (leader_to_close_dict[vehicle_id] or future_conflict_dict[vehicle_id])]
[not make_space_dict[vehicle_id] and (leader_to_close_dict[vehicle_id] or future_conflict_dict[vehicle_id])]
[make_space_dict[vehicle_id]]
Zustand 0: Normalzustand
Fahre auf die rechte Sublane und bremse ab
Zustand 1: RG warten
Fahre wieder auf der linken Sublane
[not make_space_dict[vehicle_id] and not leader_to_close_dict[vehicle_id] and not future_conflict_dict[vehicle_id]]
[not make_space_dict[vehicle_id] and not leader_to_close_dict[vehicle_id] and not future_conflict_dict[vehicle_id]]
[make_space_dict[vehicle_id]]

Backup

- Initiale Zeitplanung
- Aufbau dynamische Rettungsgasse
- Workflow der Programmnutzung
- Diagramme
- Zustandsautomat
- Equilibrierung
- Weitere Ergebnisse
- GUI Screenshots

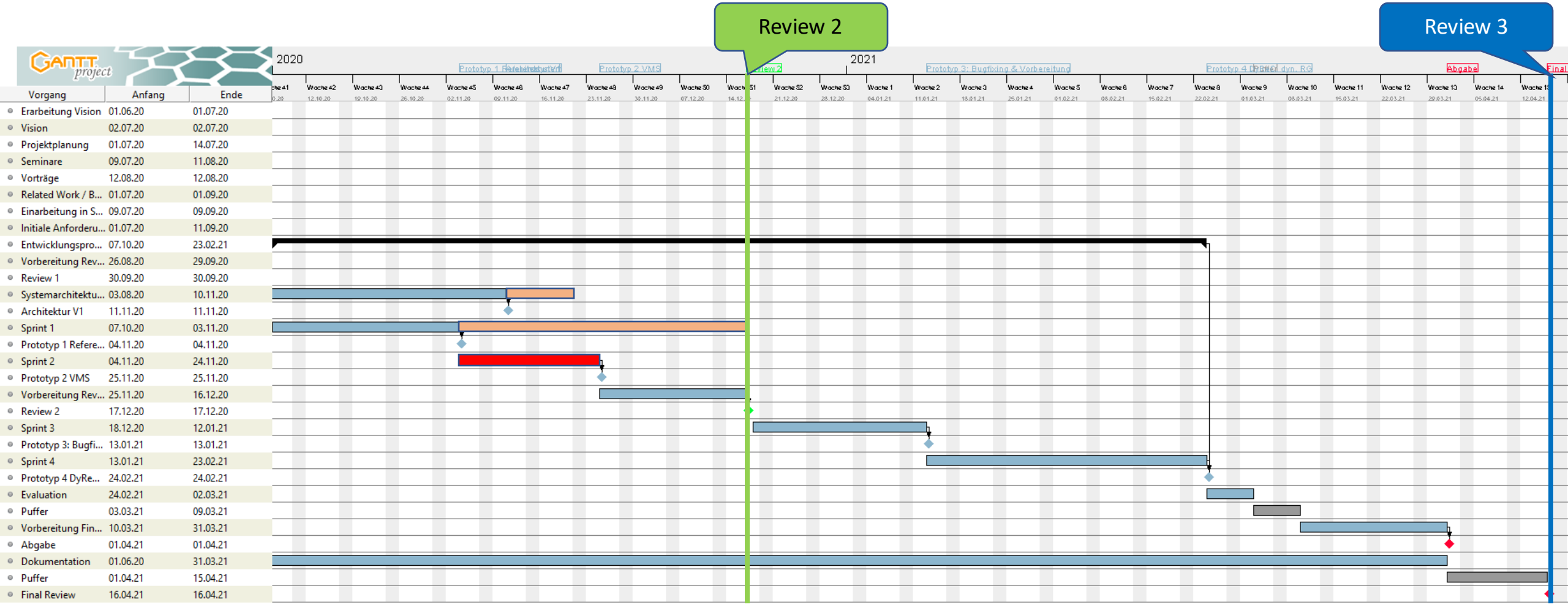


Vorgang	Anfang	Ende
Erarbeitung Vision	01.06.20	01.07.20
Vision	02.07.20	02.07.20
Projektplanung	01.07.20	14.07.20
Seminare	09.07.20	11.08.20
Vorträge	12.08.20	12.08.20
Related Work / Benchmarking	01.07.20	01.09.20
Einarbeitung in Sumo	09.07.20	09.09.20
Initiale Anforderungsanalyse	01.07.20	11.09.20
Entwicklungsprozess nach Scrum	07.10.20	16.02.21
Vorbereitung Review	26.08.20	29.09.20
Review 1	30.09.20	30.09.20
Systemarchitektur erarbeiten	03.08.20	15.10.20
Architektur V1	16.10.20	16.10.20
Sprint 1	07.10.20	03.11.20
Prototyp 1 Referenzsystem	04.11.20	04.11.20
Sprint 2	04.11.20	24.11.20
Prototyp 2 VMS	25.11.20	25.11.20
Vorbereitung Review 2	25.11.20	16.12.20
Review 2	17.12.20	17.12.20
Sprint 3	18.12.20	19.01.21
Prototyp 3 DyReC I: kask. V. + A.	20.01.21	20.01.21
Sprint 4	20.01.21	16.02.21
Prototyp 4 DyReC II: dyn. RG	17.02.21	17.02.21
Evaluation	17.02.21	23.02.21
Puffer	24.02.21	09.03.21
Vorbereitung Final Review	10.03.21	31.03.21
Abgabe	01.04.21	01.04.21
Dokumentation	01.06.20	31.03.21
Puffer	01.04.21	15.04.21
Final Review	16.04.21	16.04.21



geplant verlängert oder verschoben gestrichen

Rückblick Zeitplanung zum Review 2



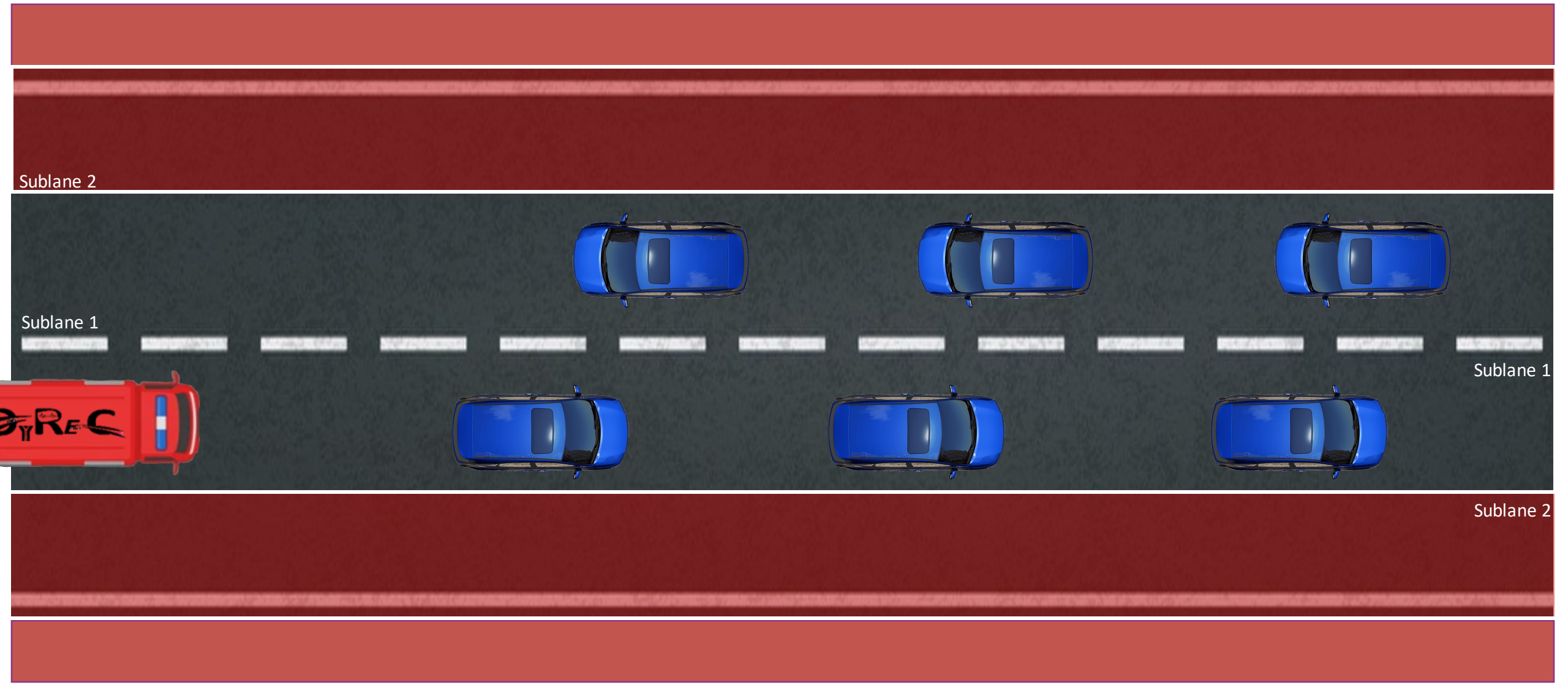
Vorgang	Anfang	Ende
Erarbeitung Vision	01.06.20	01.07.20
Vision	02.07.20	02.07.20
Projektplanung	01.07.20	14.07.20
Seminare	09.07.20	11.08.20
Vorträge	12.08.20	12.08.20
Related Work / B...	01.07.20	01.09.20
Einarbeitung in S...	09.07.20	09.09.20
Initiale Anforderu...	01.07.20	11.09.20
Entwicklungspro...	07.10.20	23.02.21
Vorbereitung Rev...	26.08.20	29.09.20
Review 1	30.09.20	30.09.20
Systemarchitektu...	03.08.20	10.11.20
Architektur V1	11.11.20	11.11.20
Sprint 1	07.10.20	03.11.20
Prototyp 1 Refere...	04.11.20	04.11.20
Sprint 2	04.11.20	24.11.20
Prototyp 2 VMS	25.11.20	25.11.20
Vorbereitung Rev...	25.11.20	16.12.20
Review 2	17.12.20	17.12.20
Sprint 3	18.12.20	12.01.21
Prototyp 3: Bugfi...	13.01.21	13.01.21
Sprint 4	13.01.21	23.02.21
Prototyp 4 DyRe...	24.02.21	24.02.21
Evaluation	24.02.21	02.03.21
Puffer	03.03.21	09.03.21
Vorbereitung Fin...	10.03.21	31.03.21
Abgabe	01.04.21	01.04.21
Dokumentation	01.06.20	31.03.21
Puffer	01.04.21	15.04.21
Final Review	16.04.21	16.04.21

Geplant

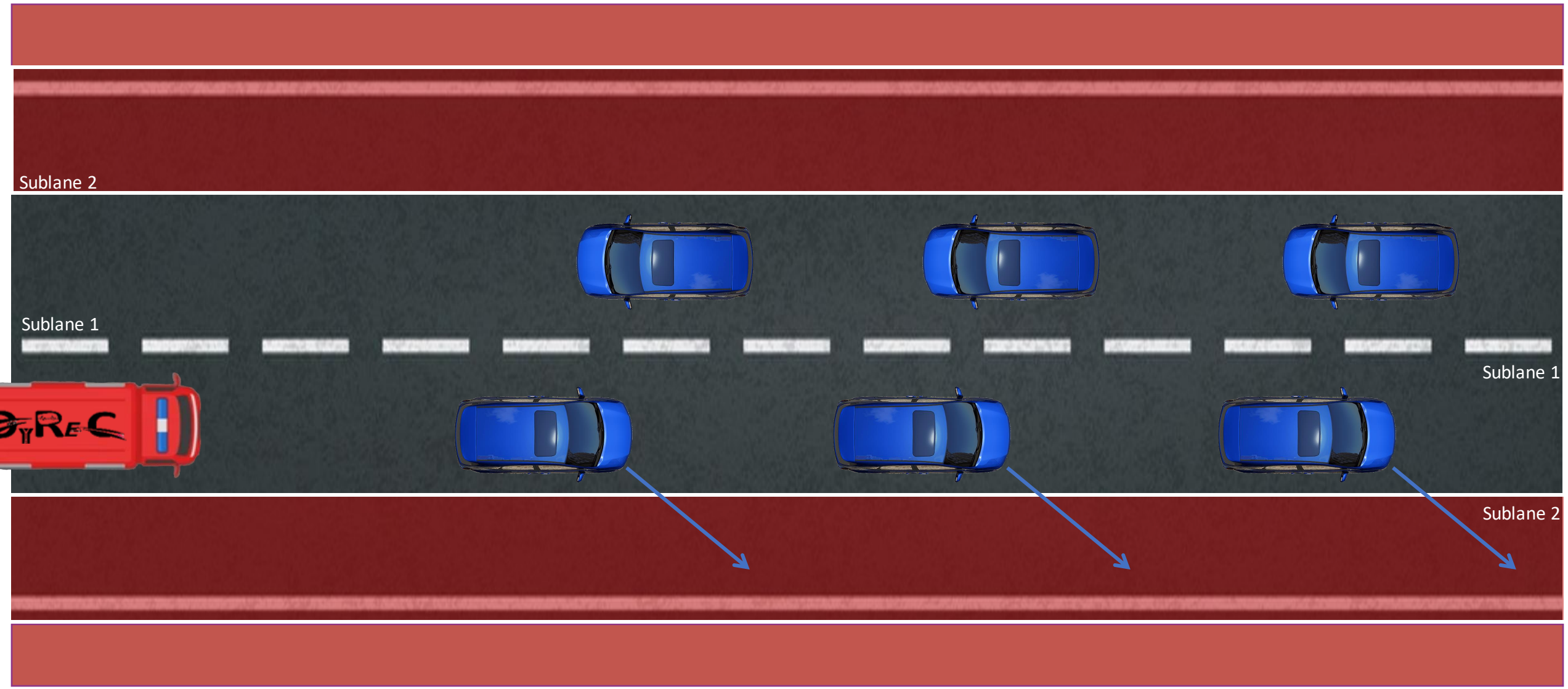
Verlängert/Verschoben

Gestrichen

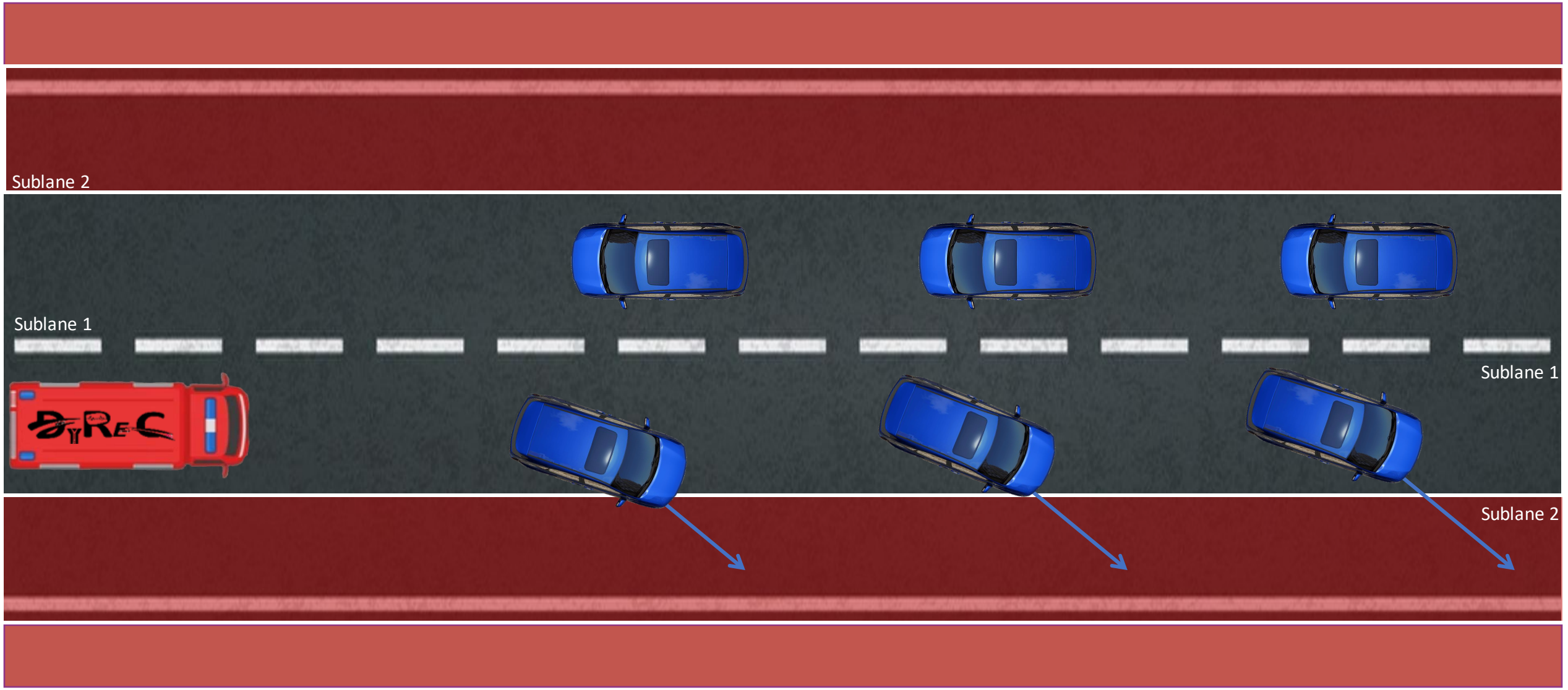
Dynamische Rettungsgasse



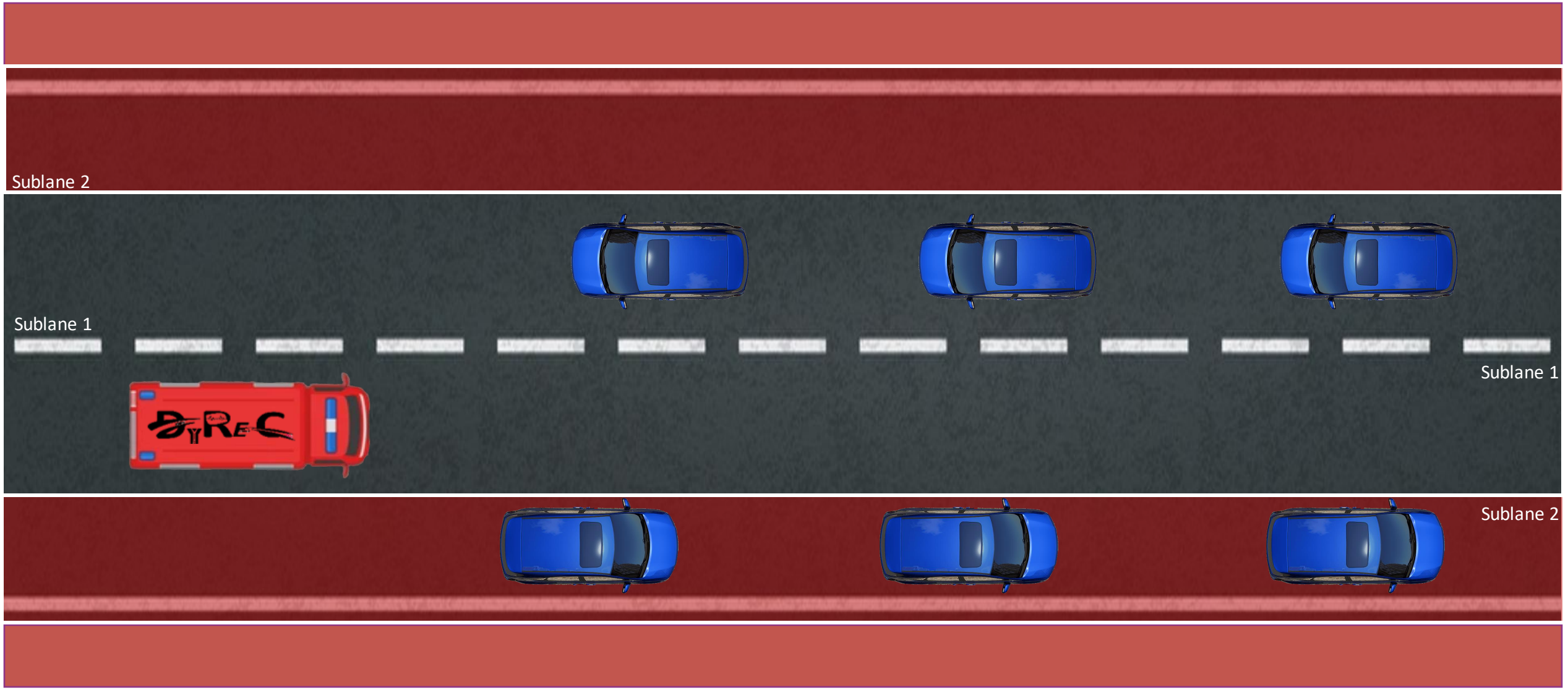
Dynamische Rettungsgasse



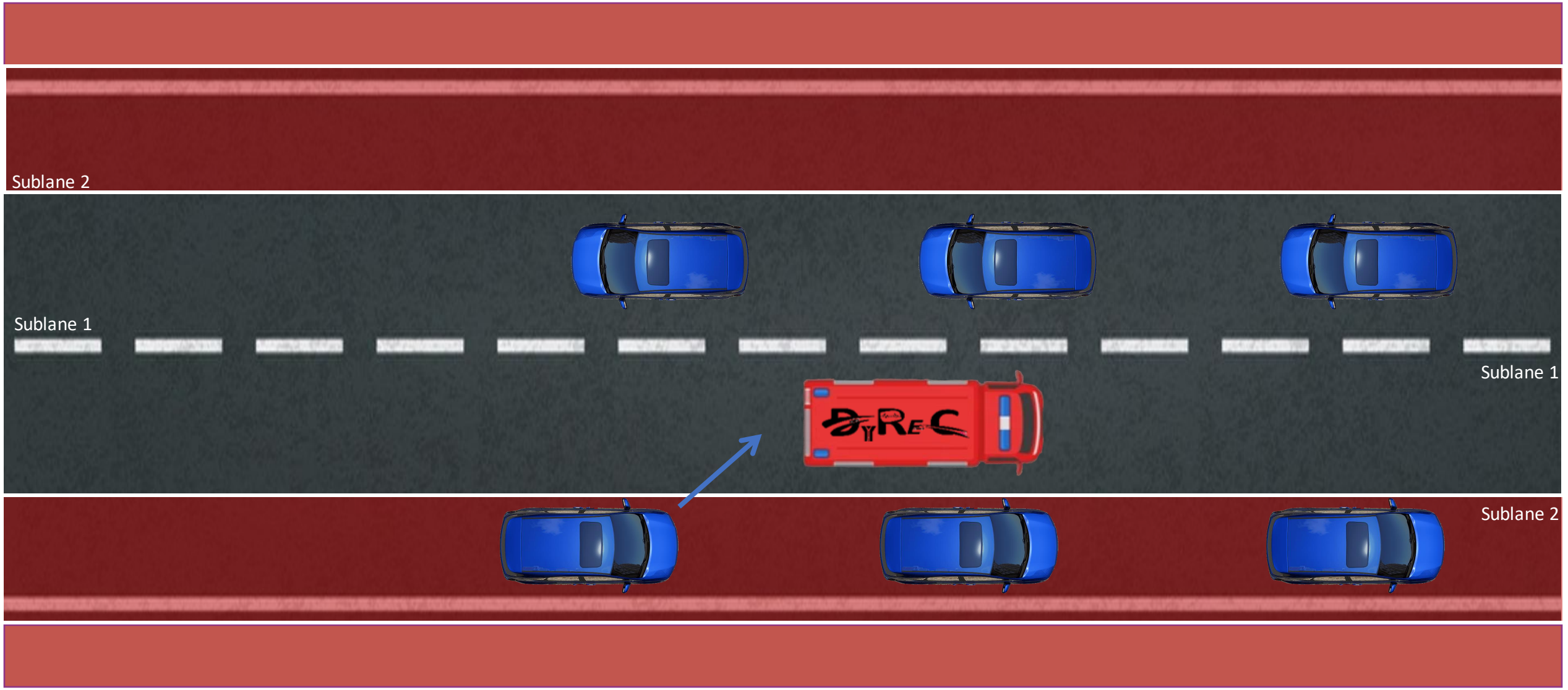
Dynamische Rettungsgasse



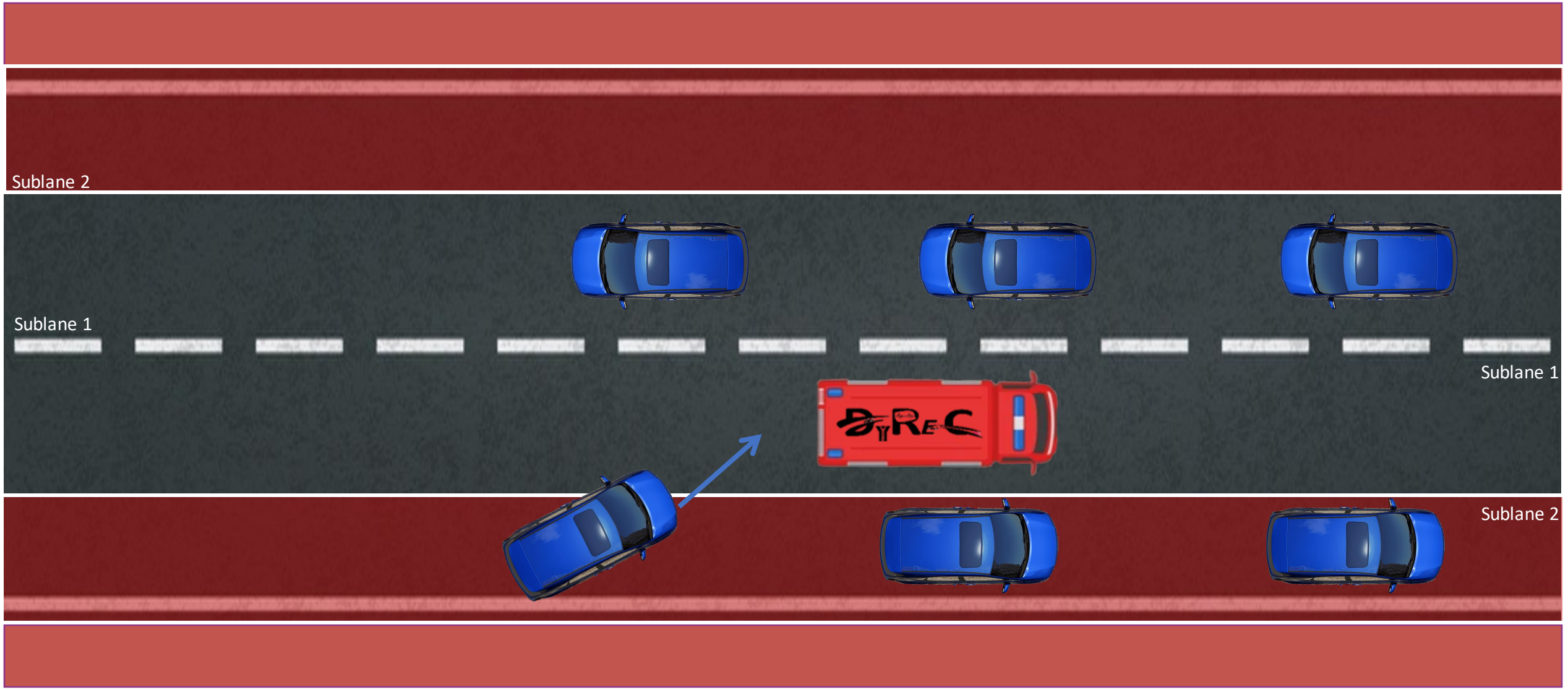
Dynamische Rettungsgasse



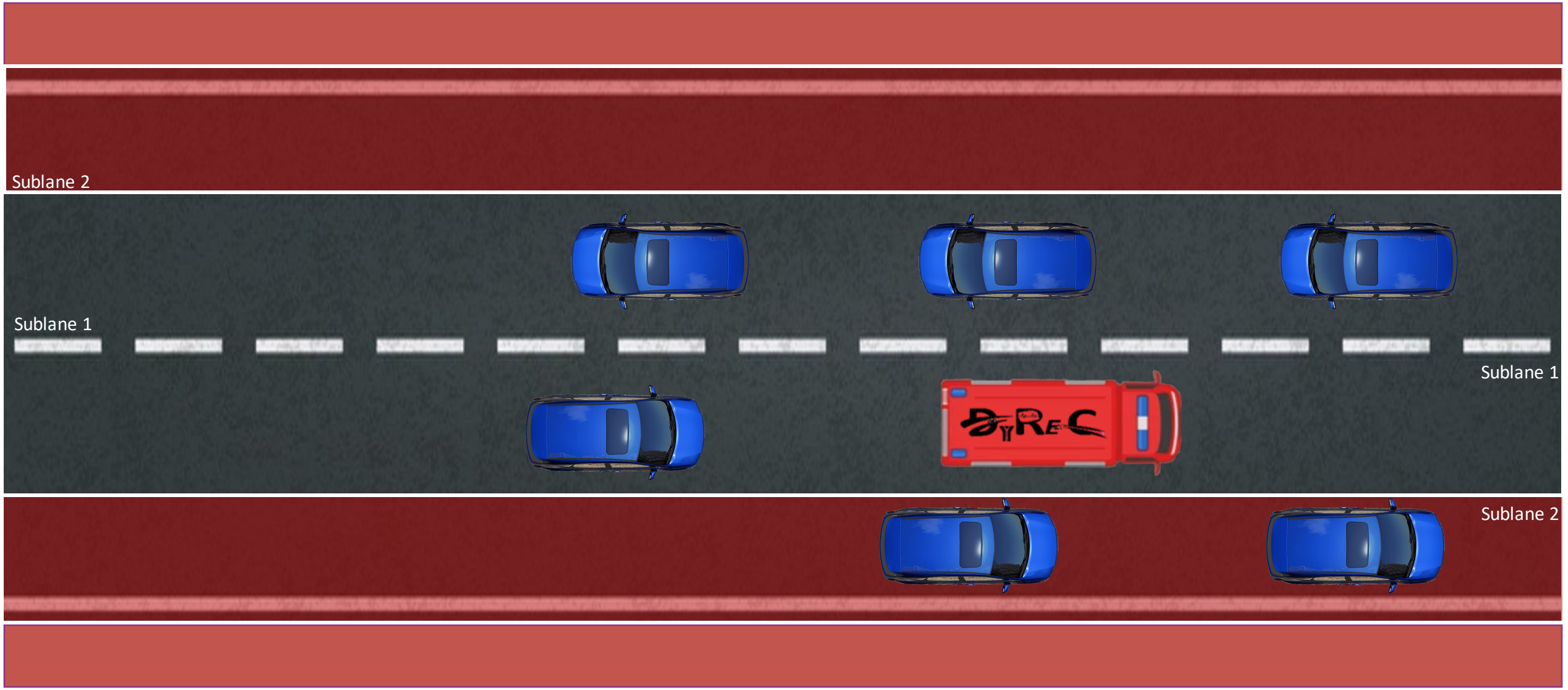
Dynamische Rettungsgasse



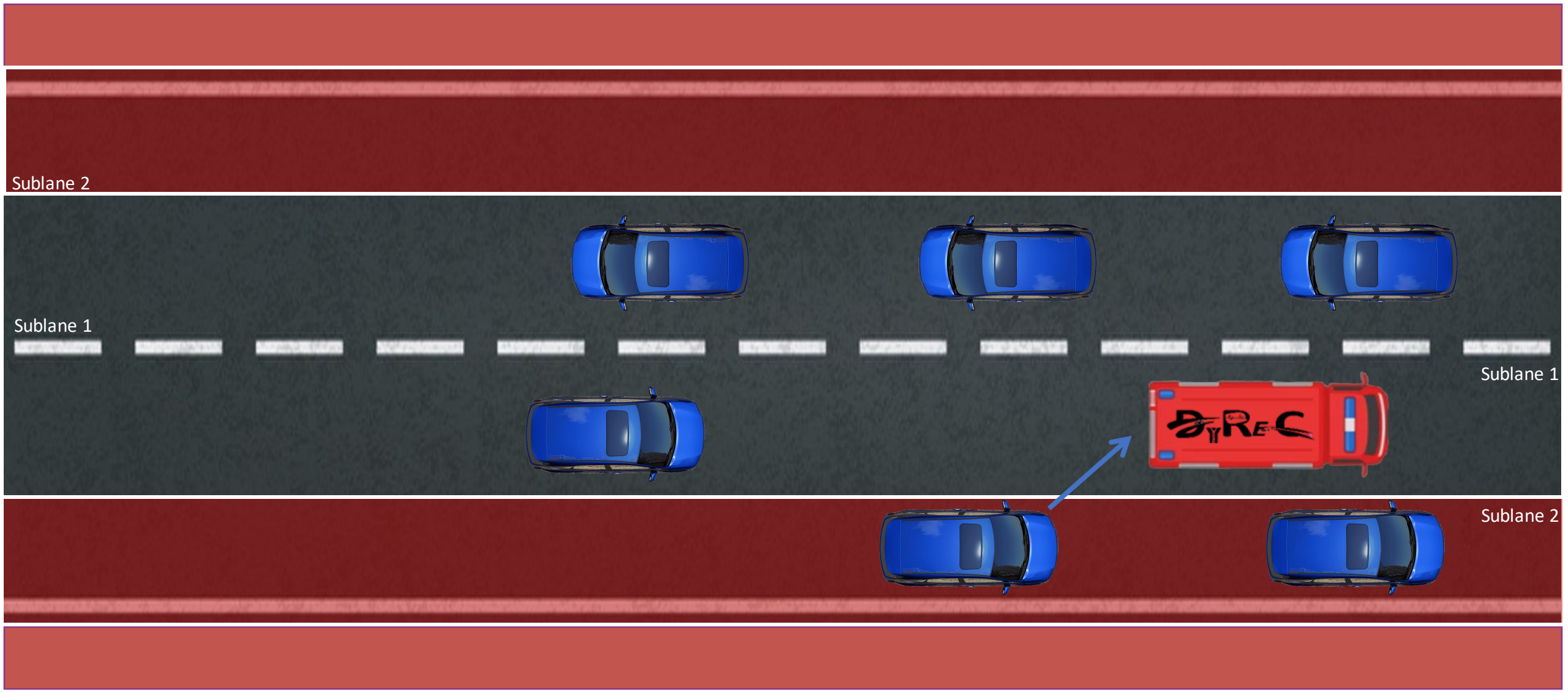
Dynamische Rettungsgasse



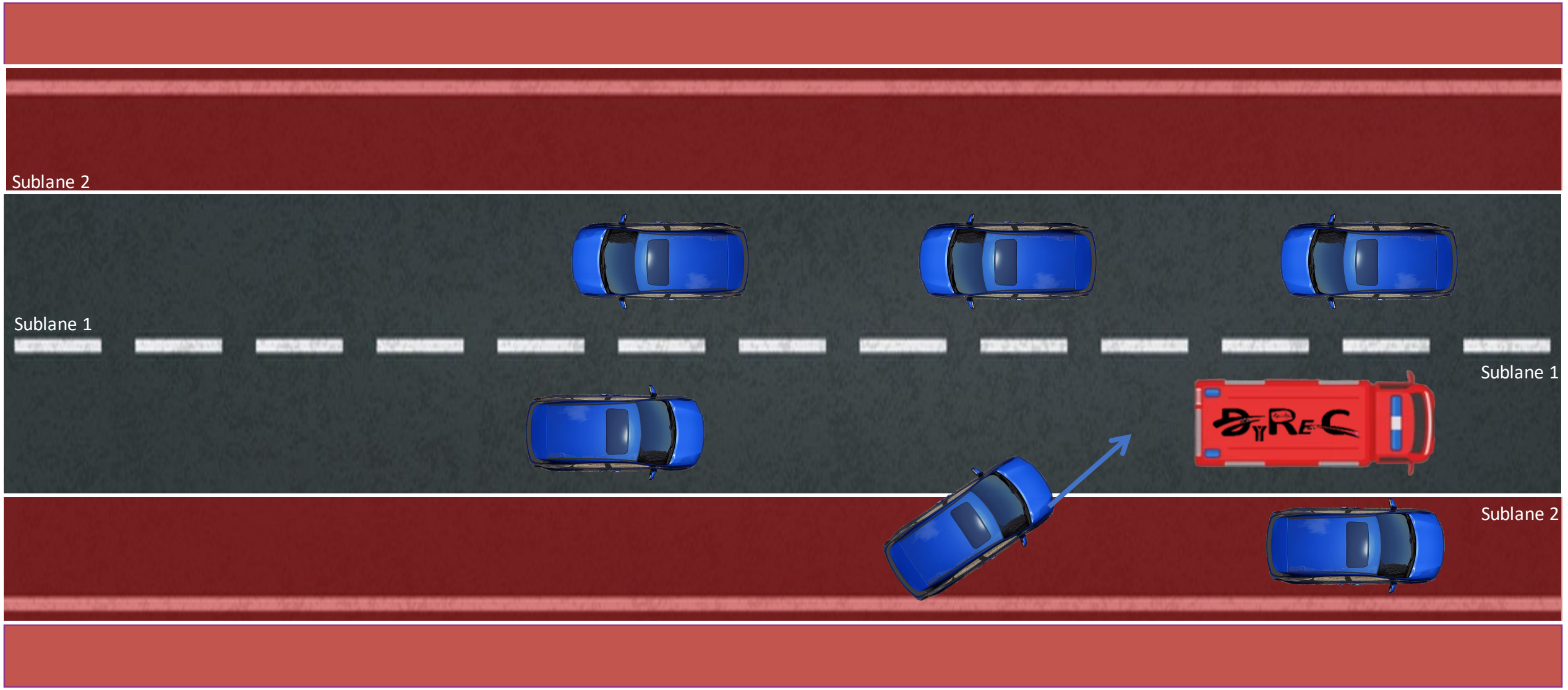
Dynamische Rettungsgasse



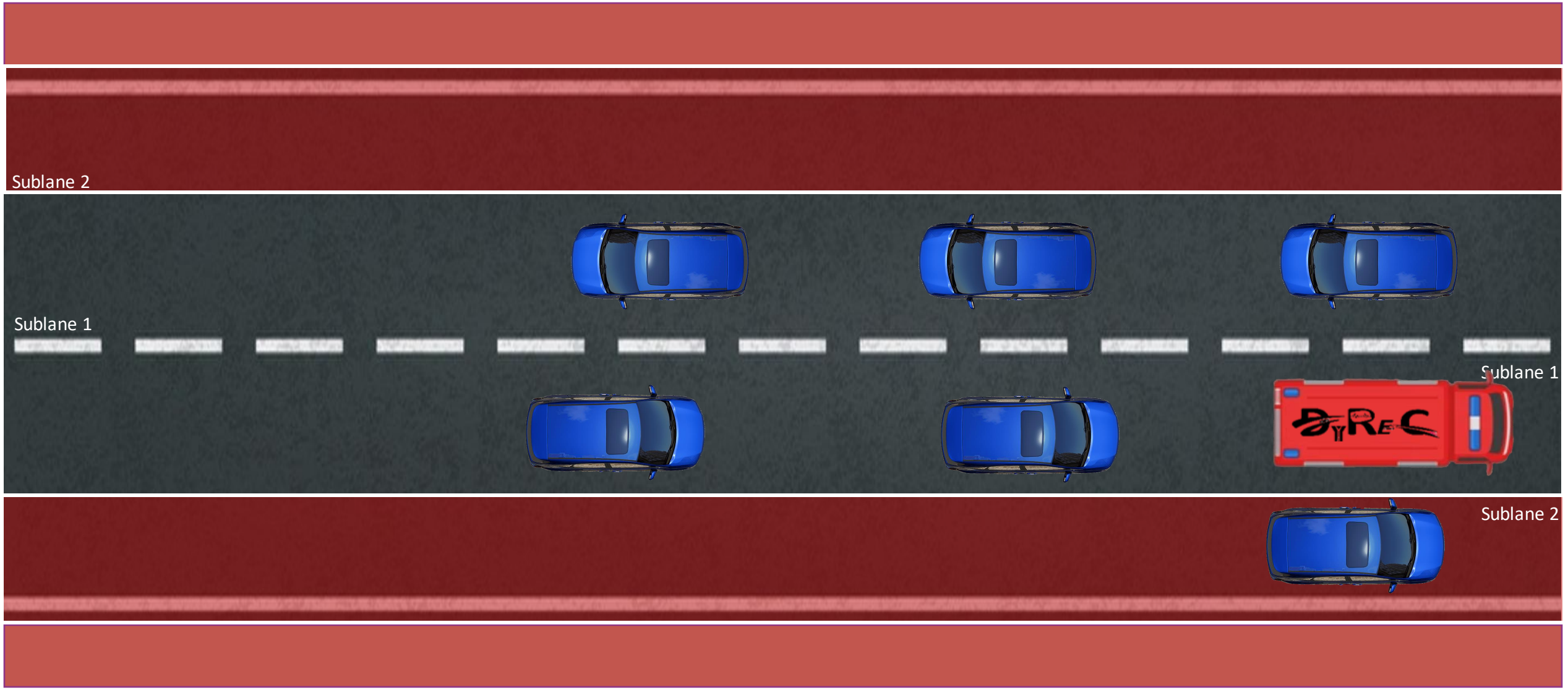
Dynamische Rettungsgasse



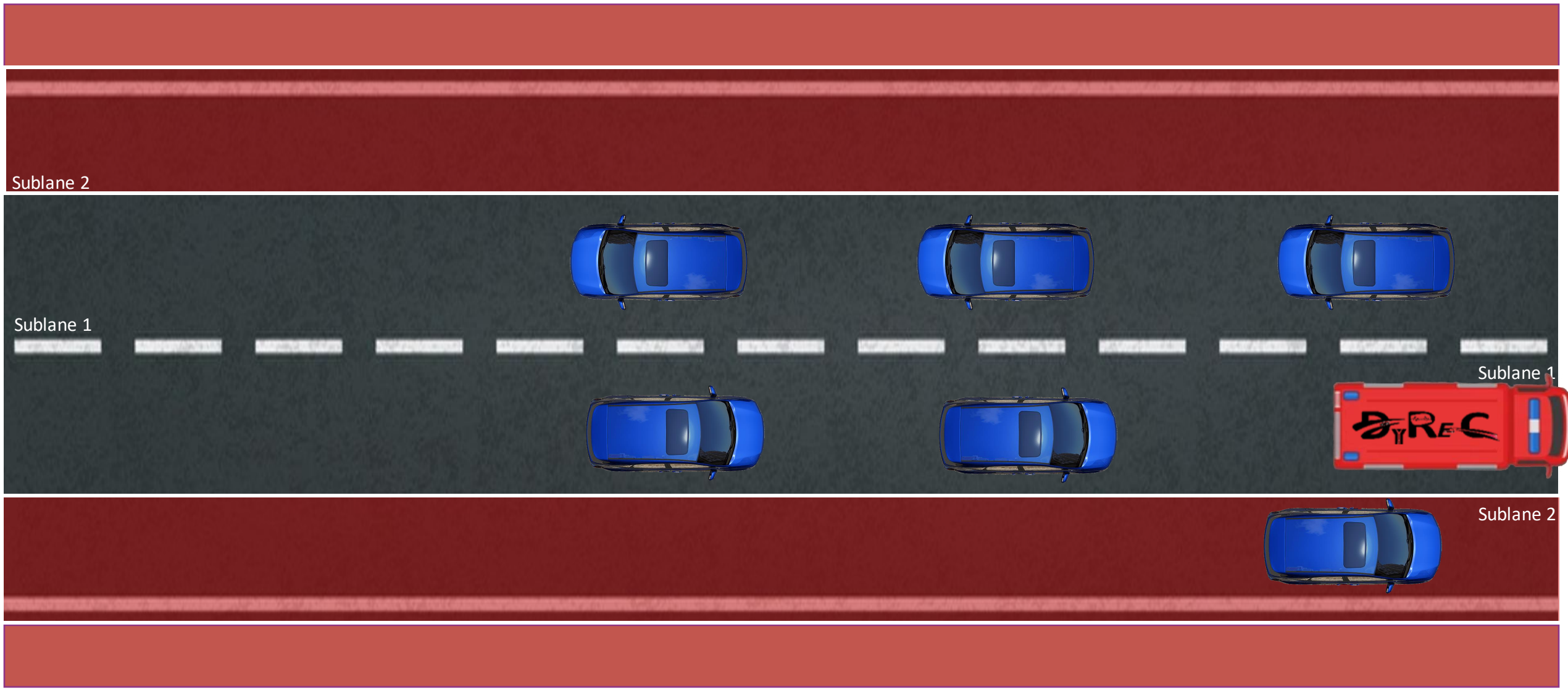
Dynamische Rettungsgasse



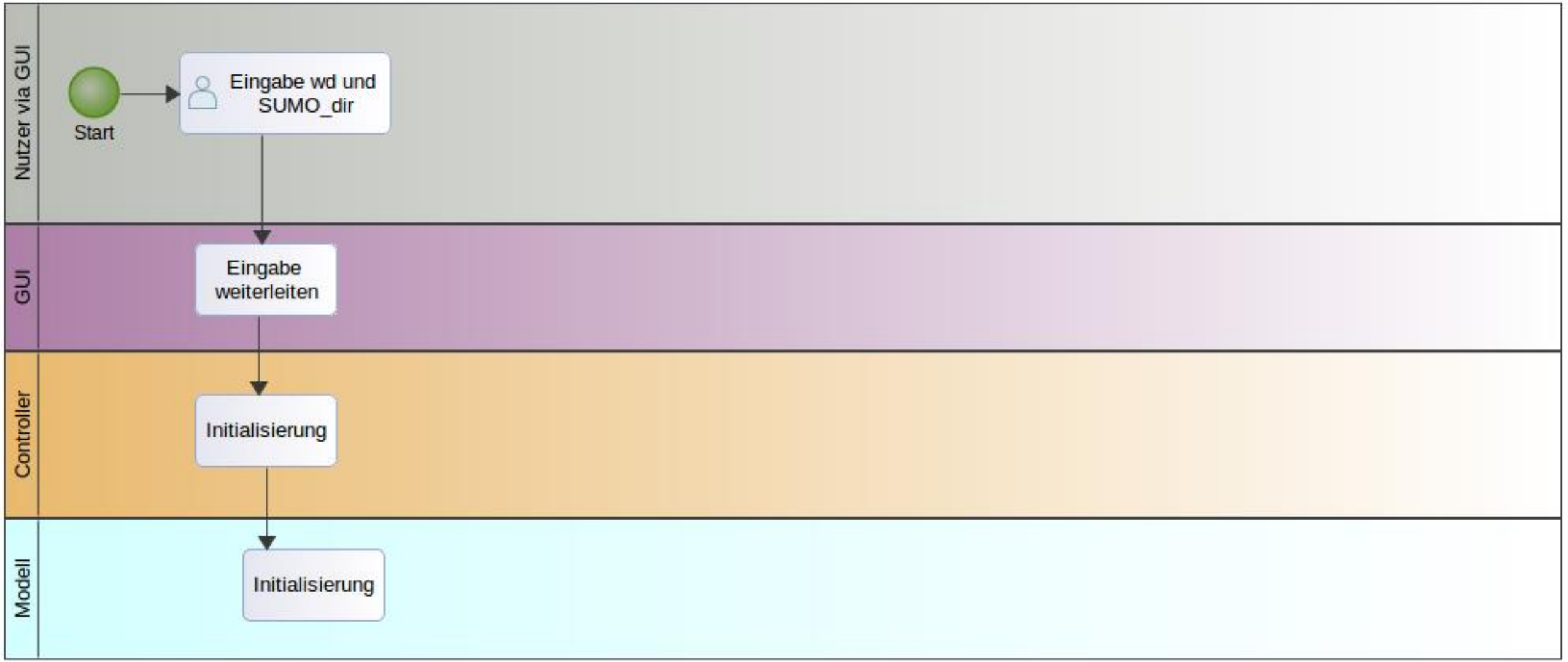
Dynamische Rettungsgasse



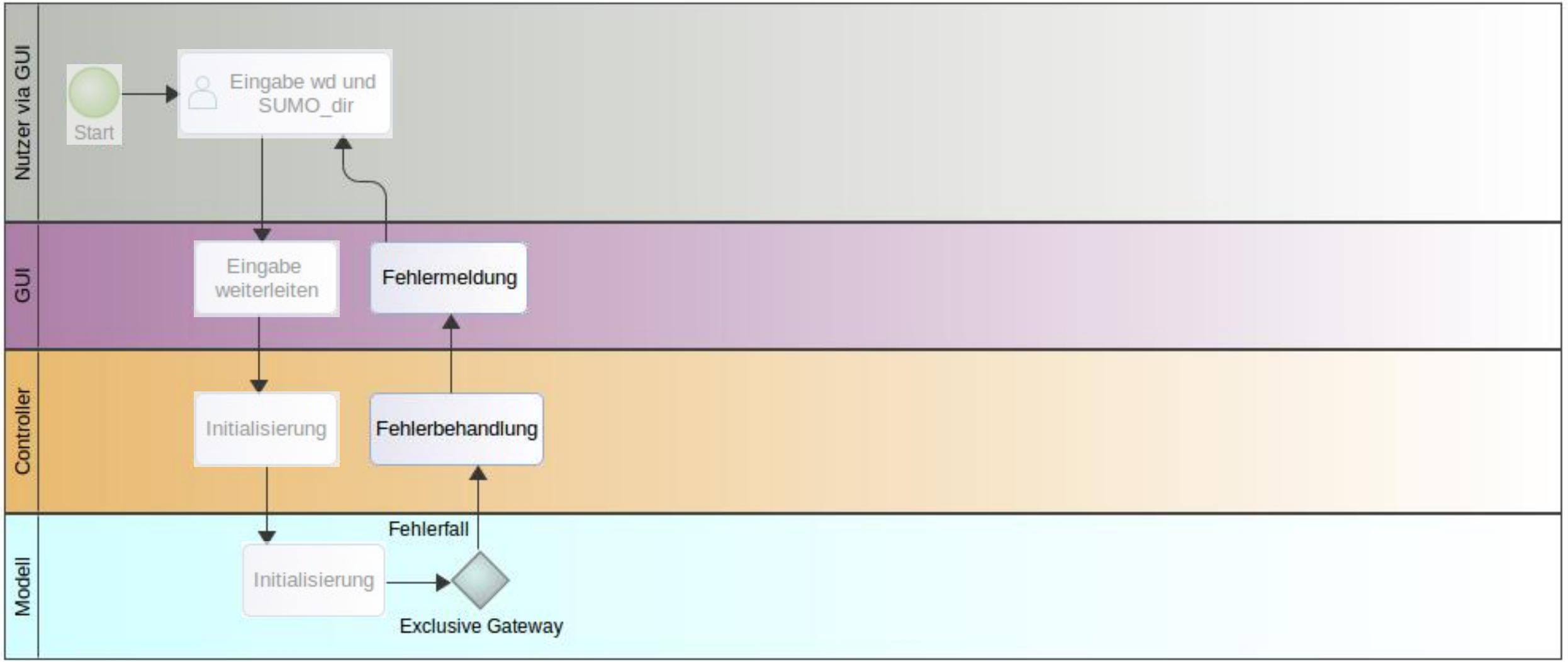
Dynamische Rettungsgasse



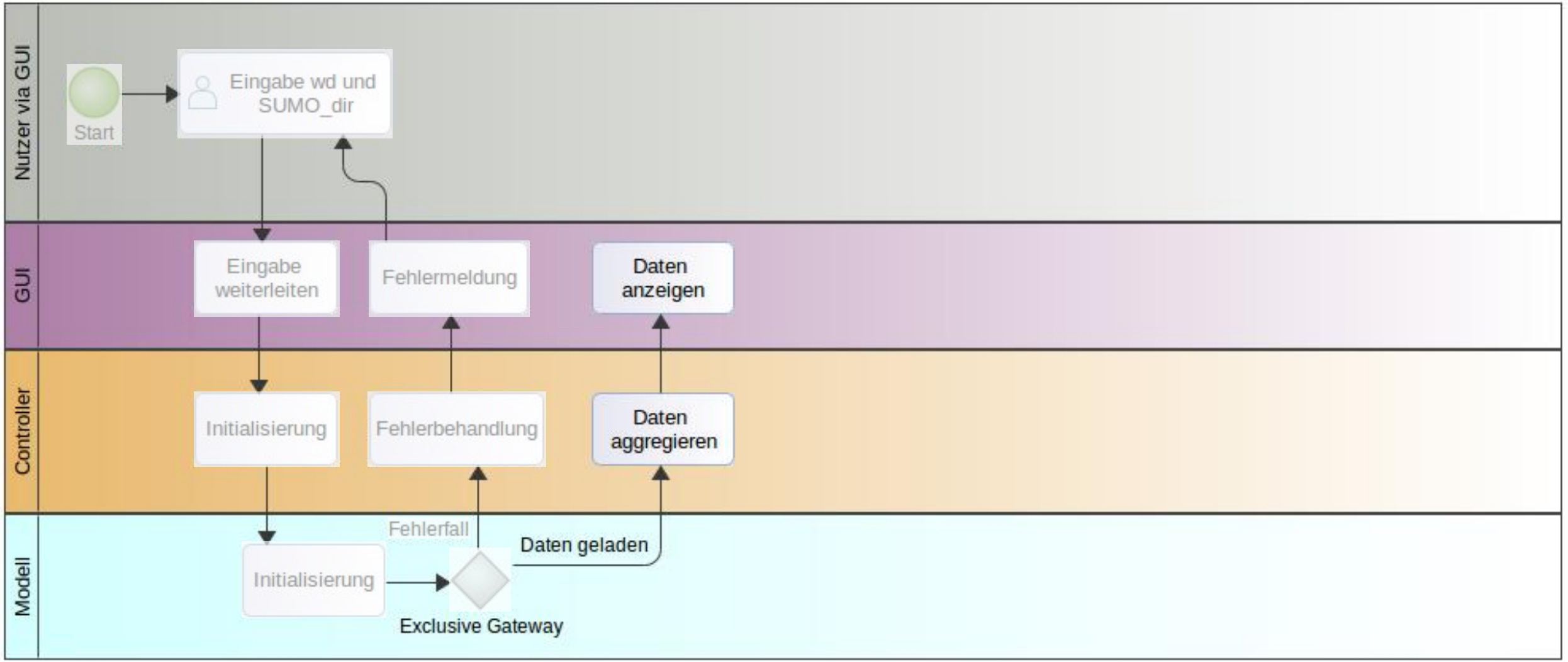
Workflow der Programmnutzung



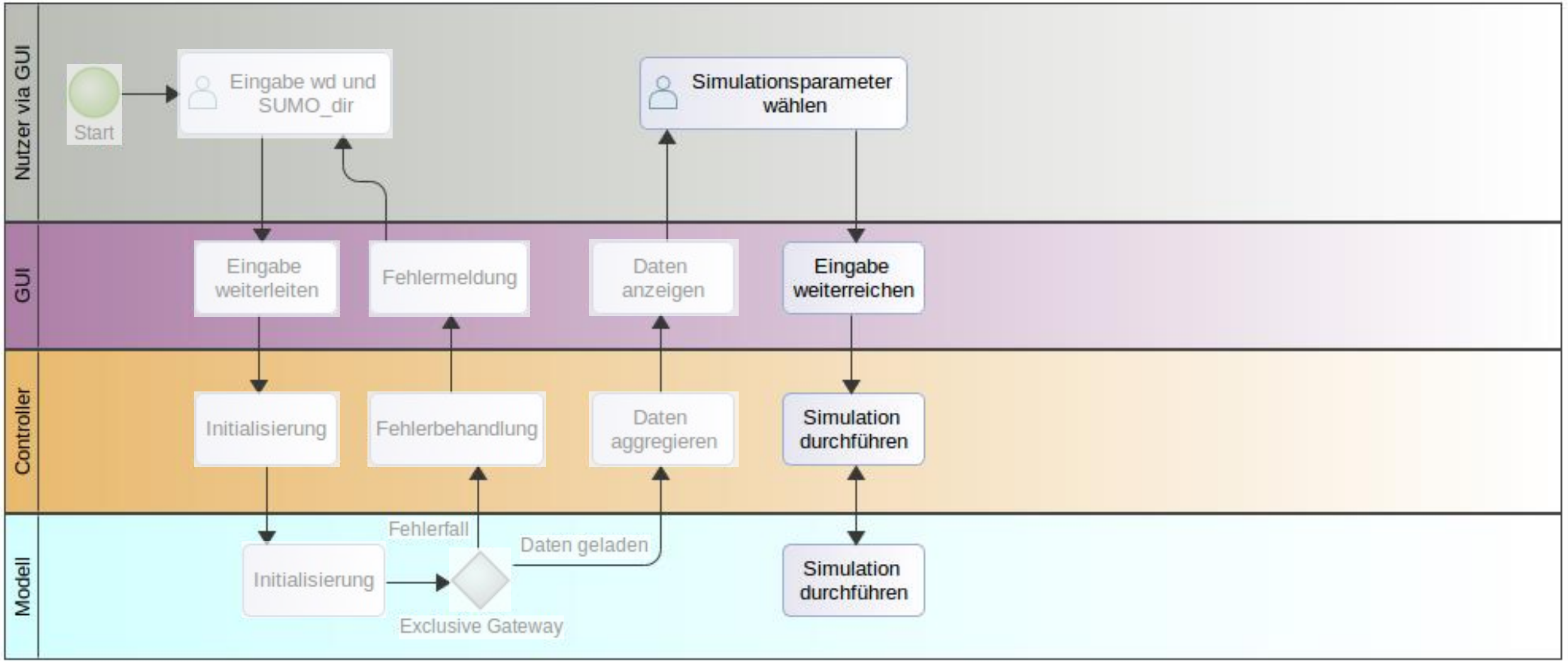
Workflow der Programmnutzung



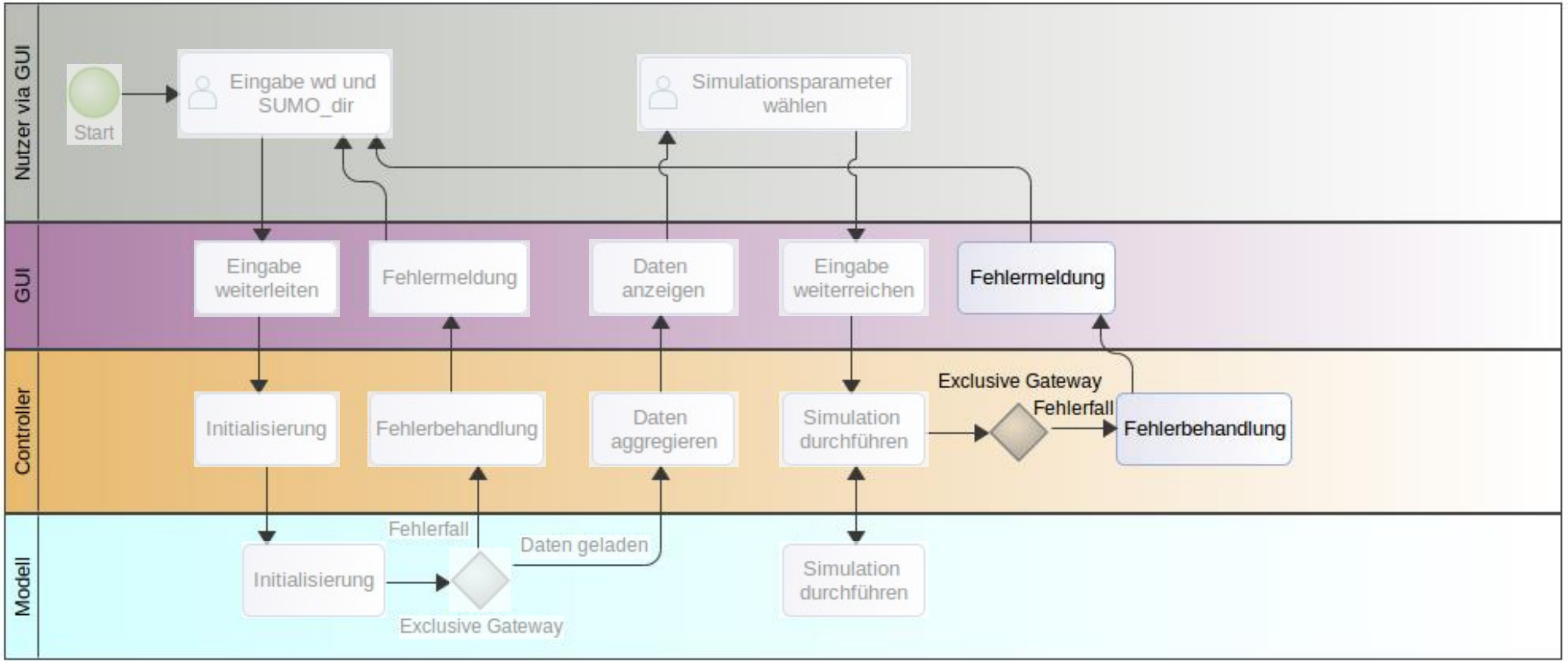
Workflow der Programmnutzung



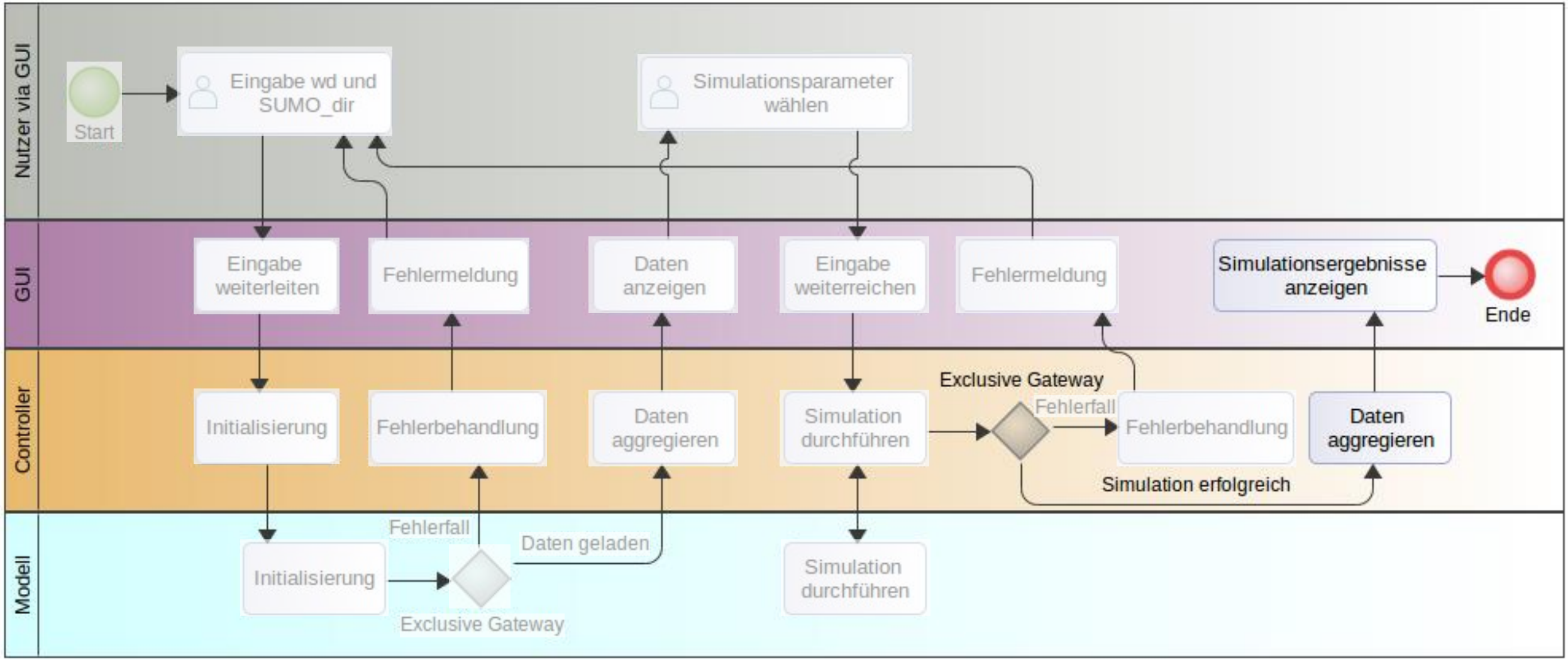
Workflow der Programmnutzung



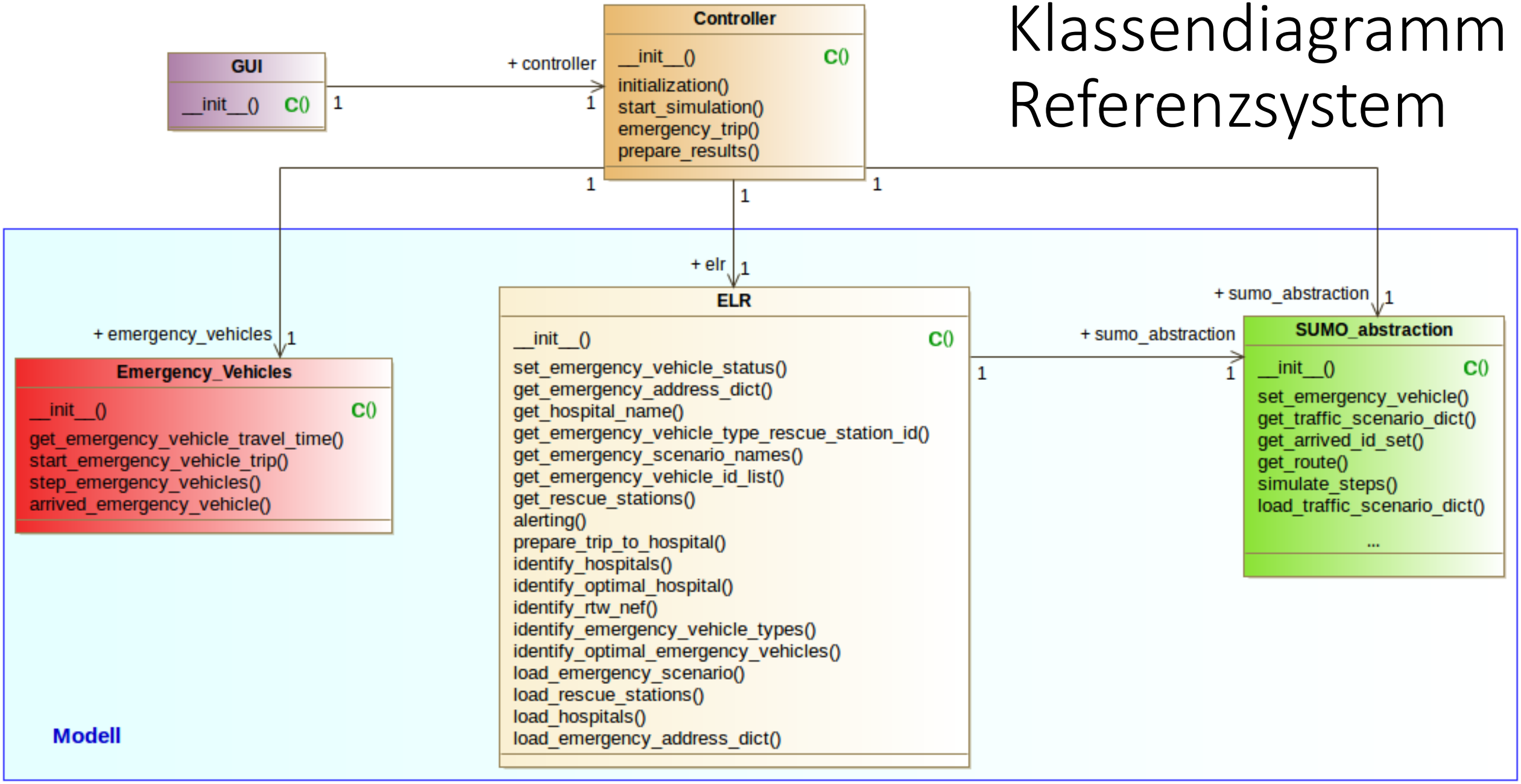
Workflow der Programmnutzung



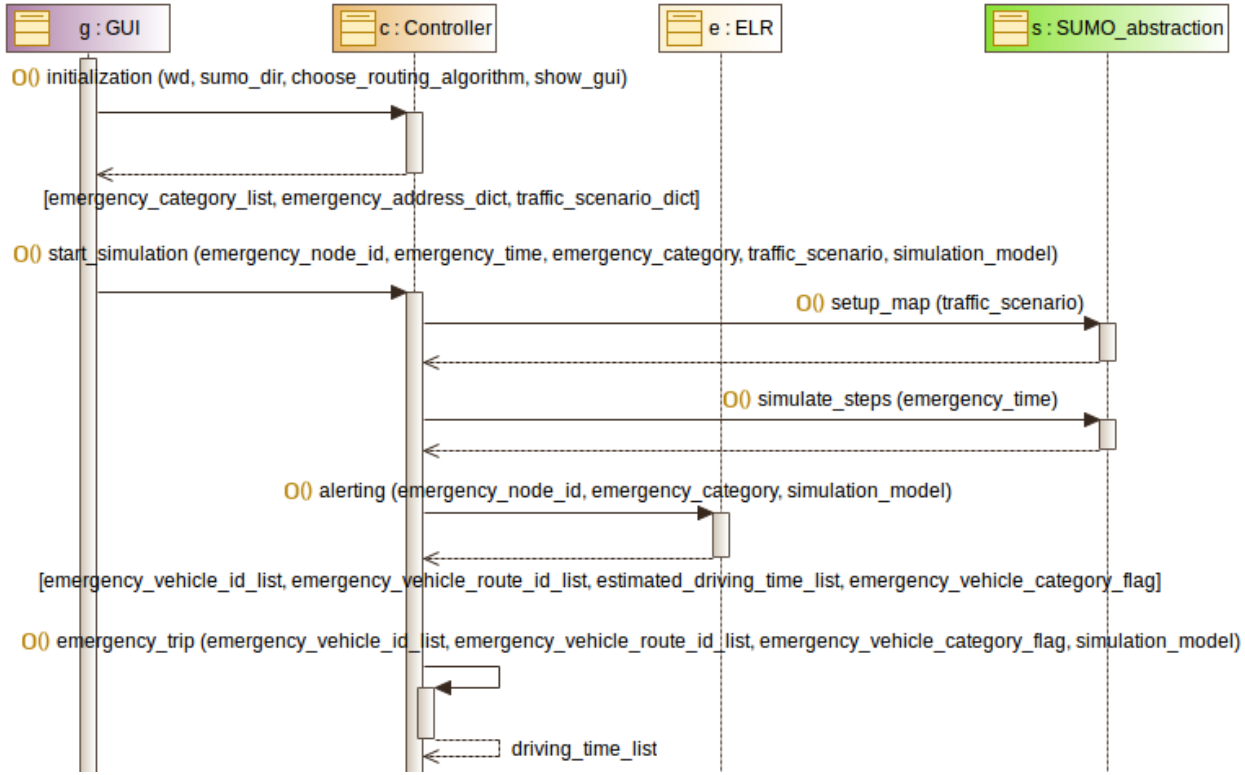
Workflow der Programmnutzung

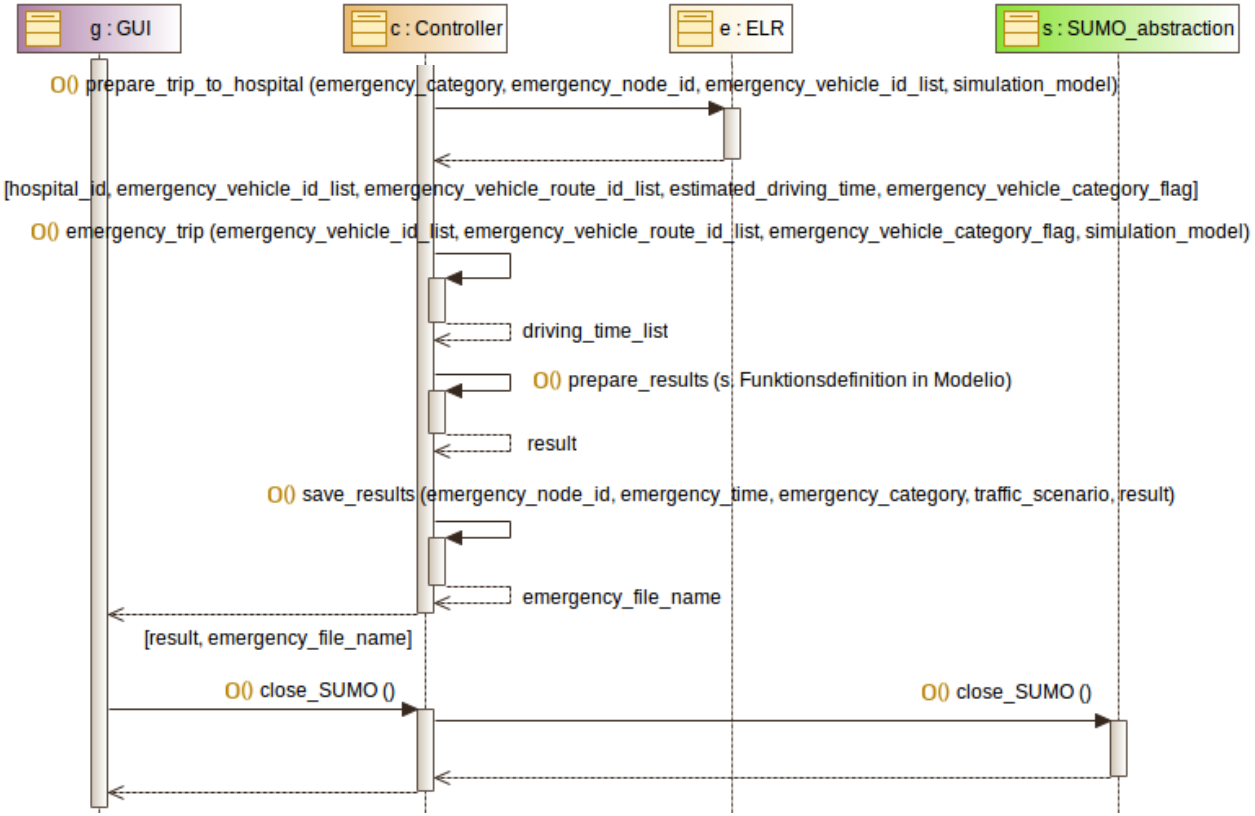


Klassendiagramm Referenzsystem

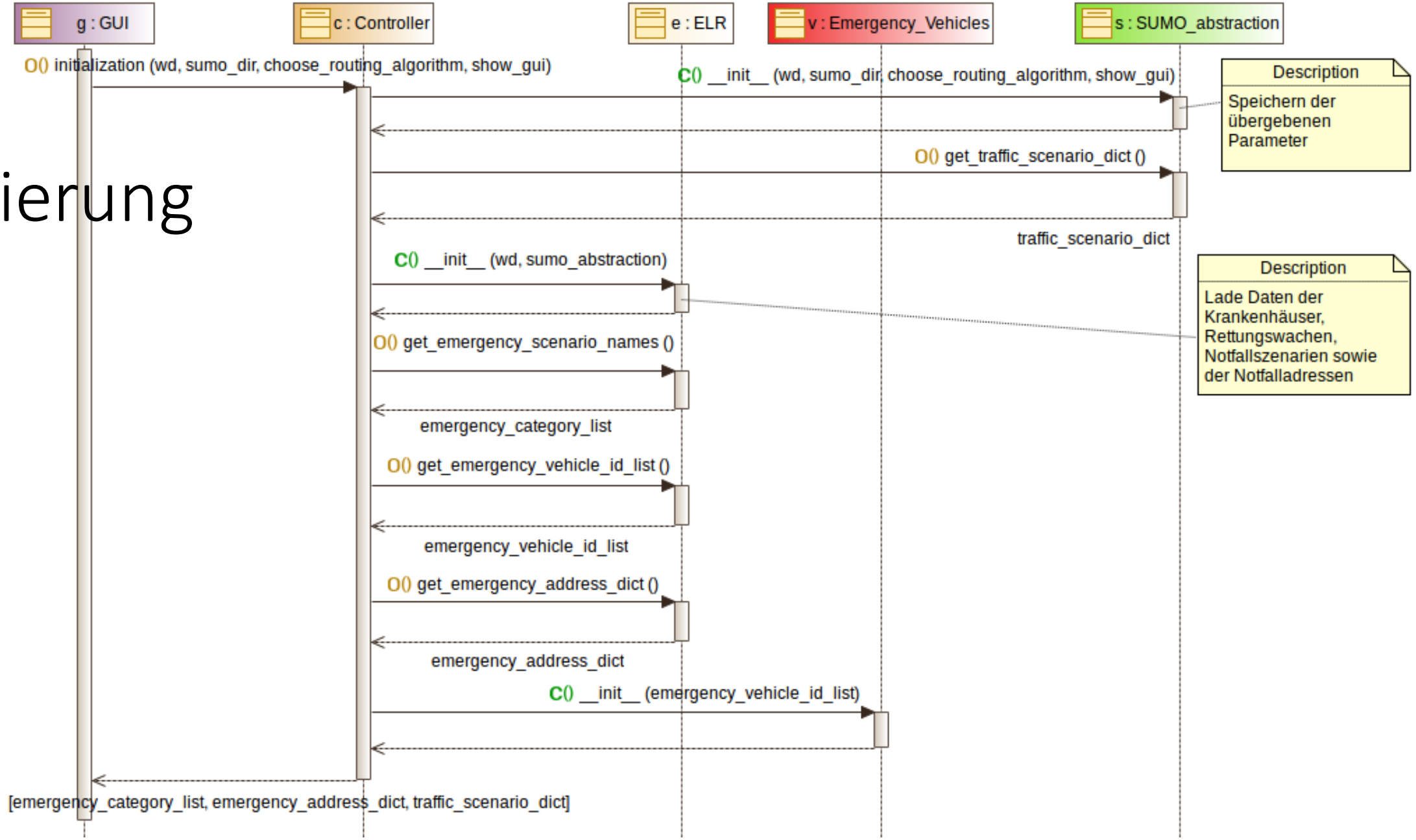


Übersicht Referenzsystem



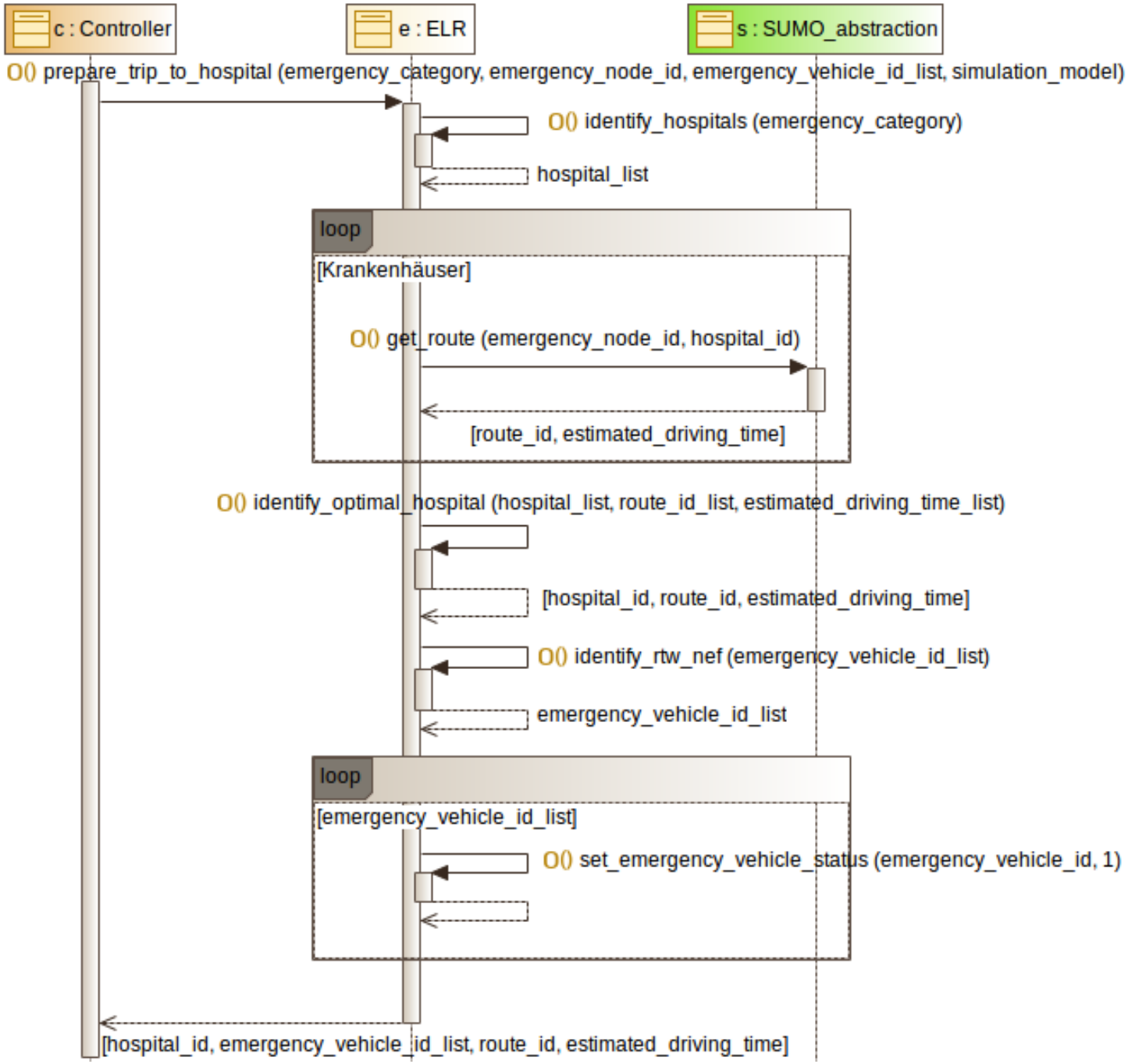


Initialisierung

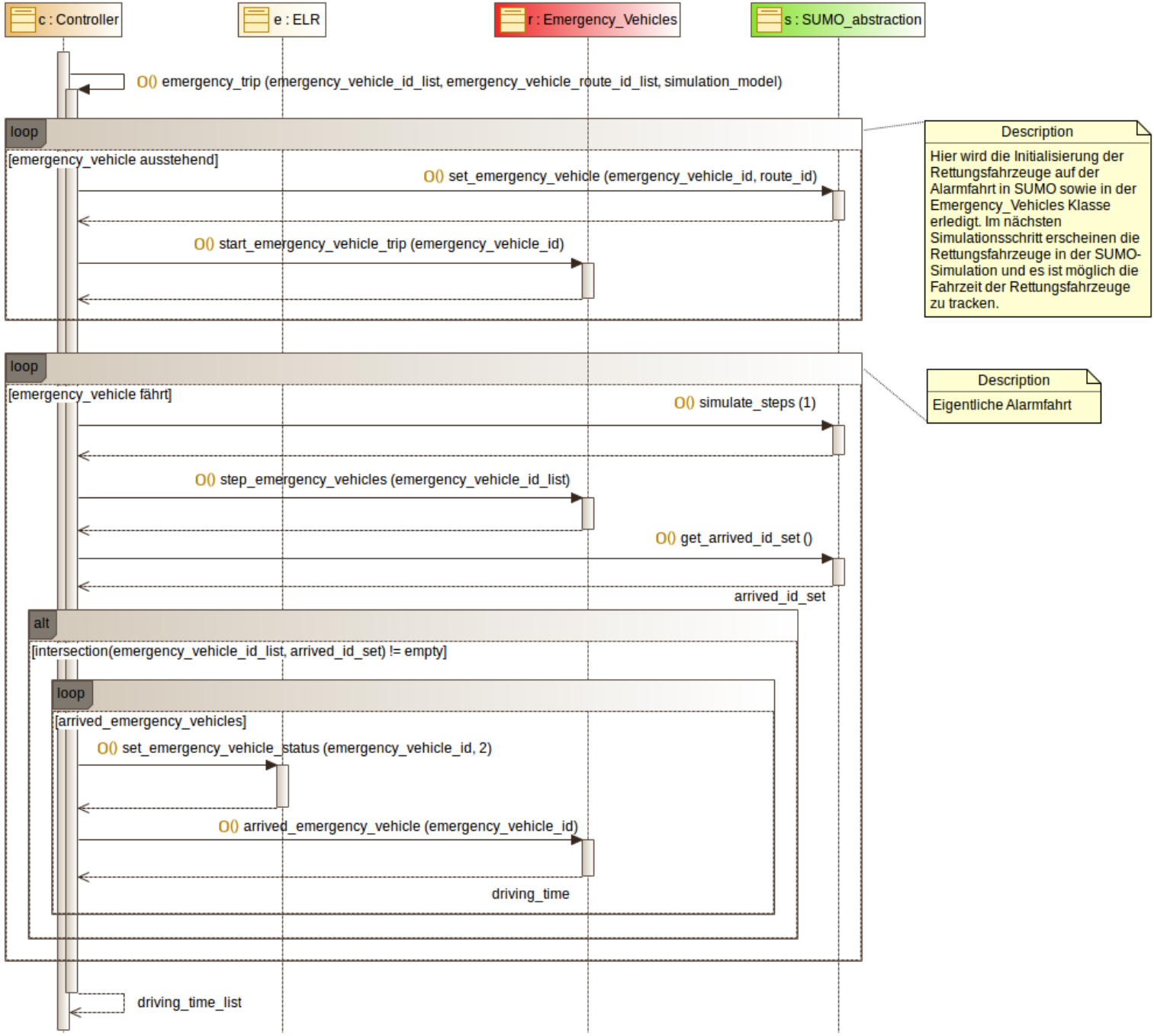


Alarmierung

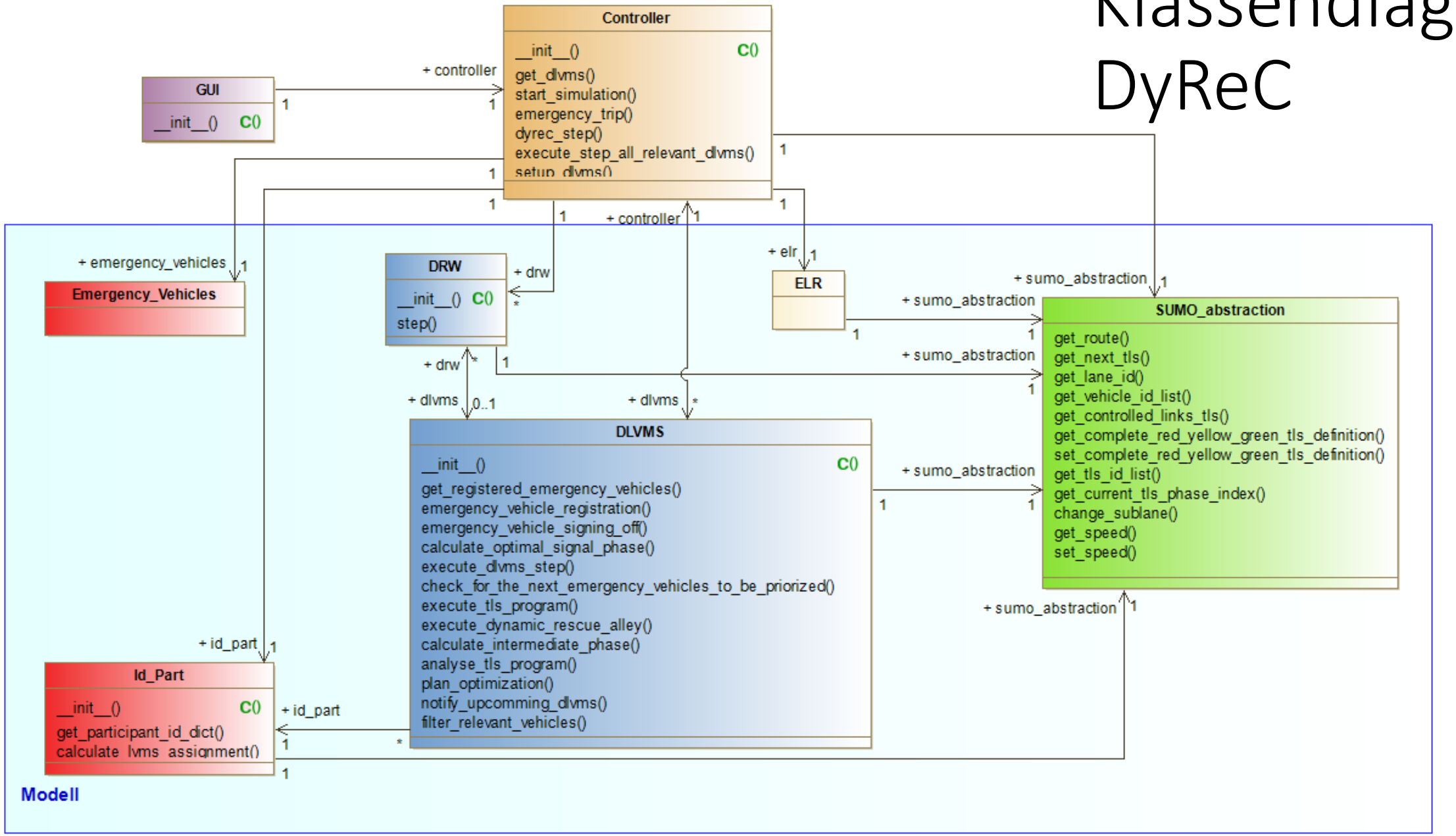
Vorbereitung der Alarmfahrt zum Krankenhaus



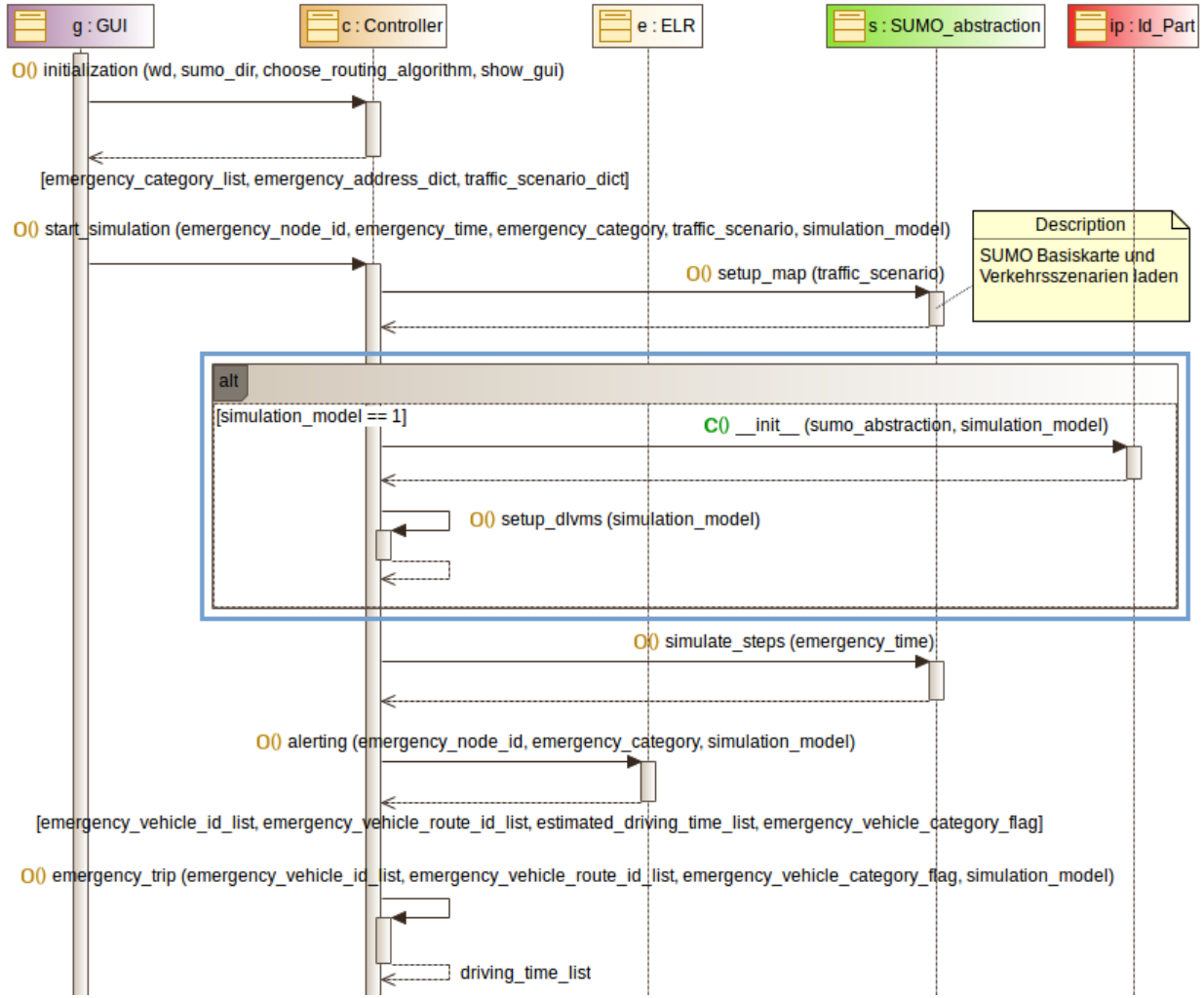
Alarmfahrt Referenzsystem

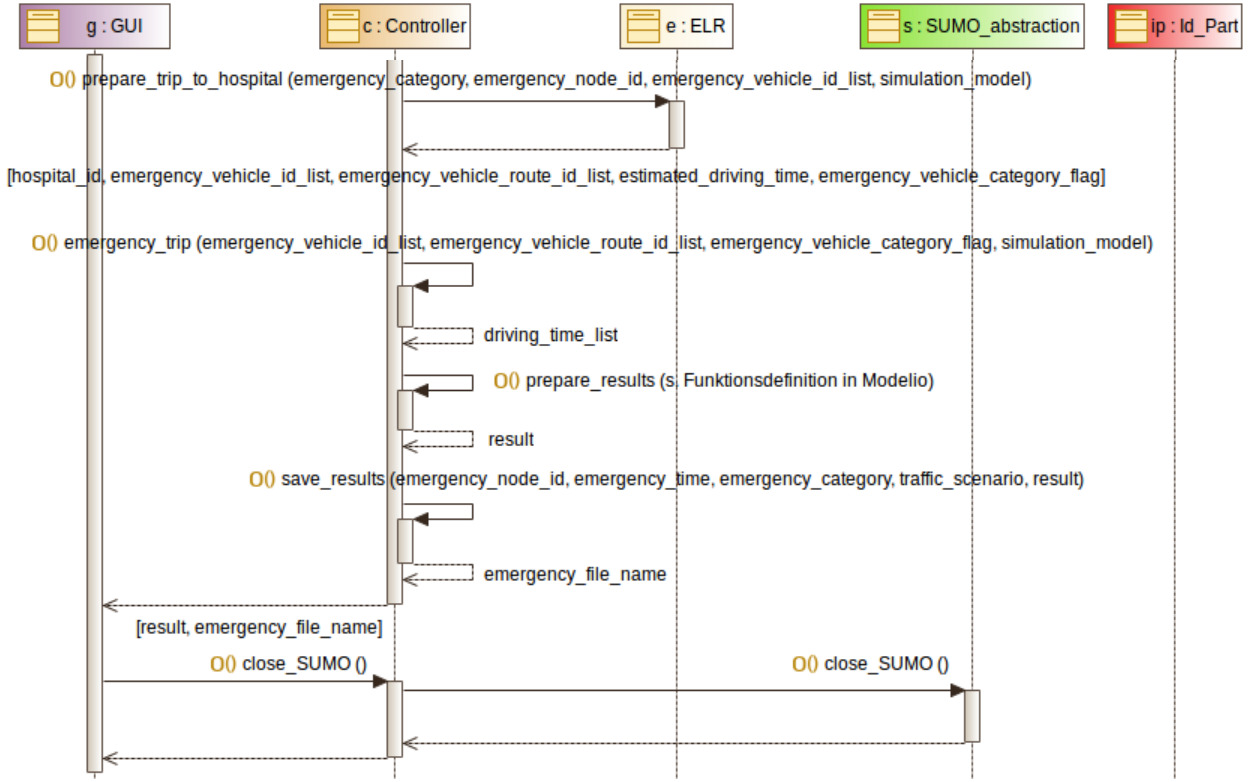


Klassendiagramm DyReC



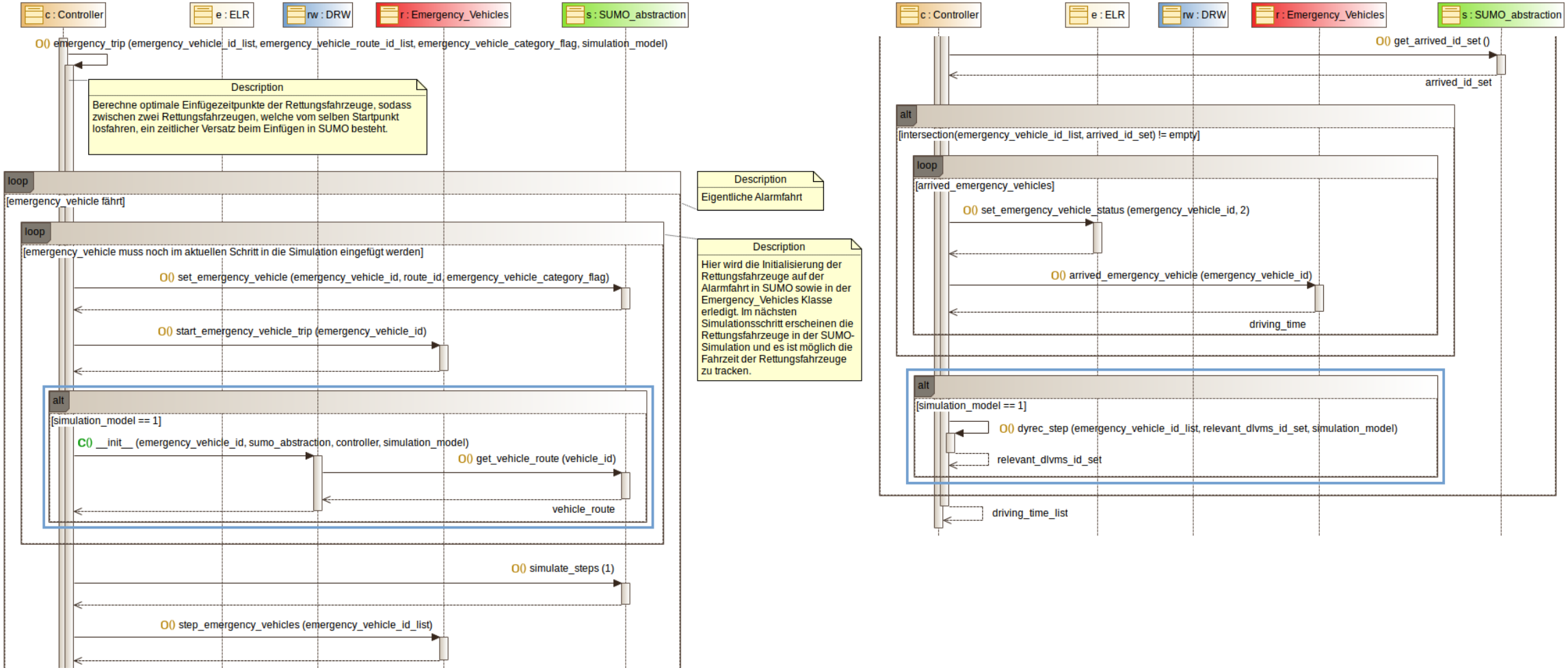
Übersicht DyReC



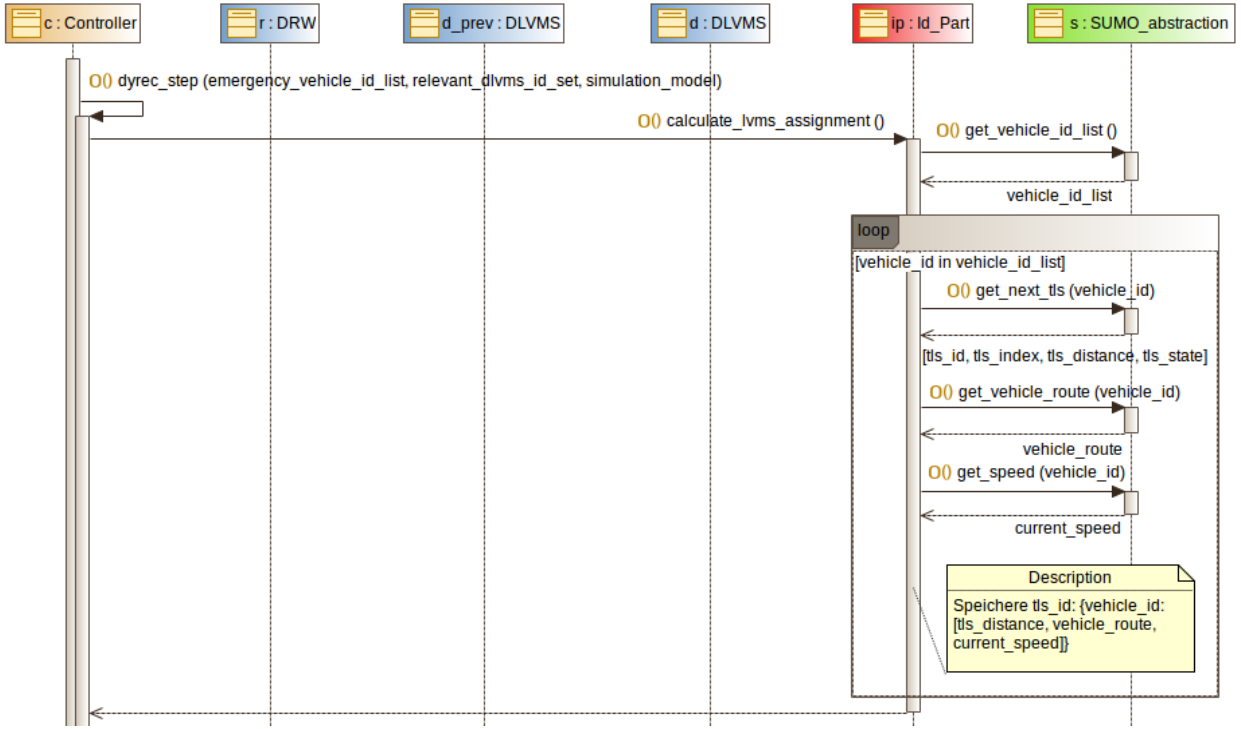


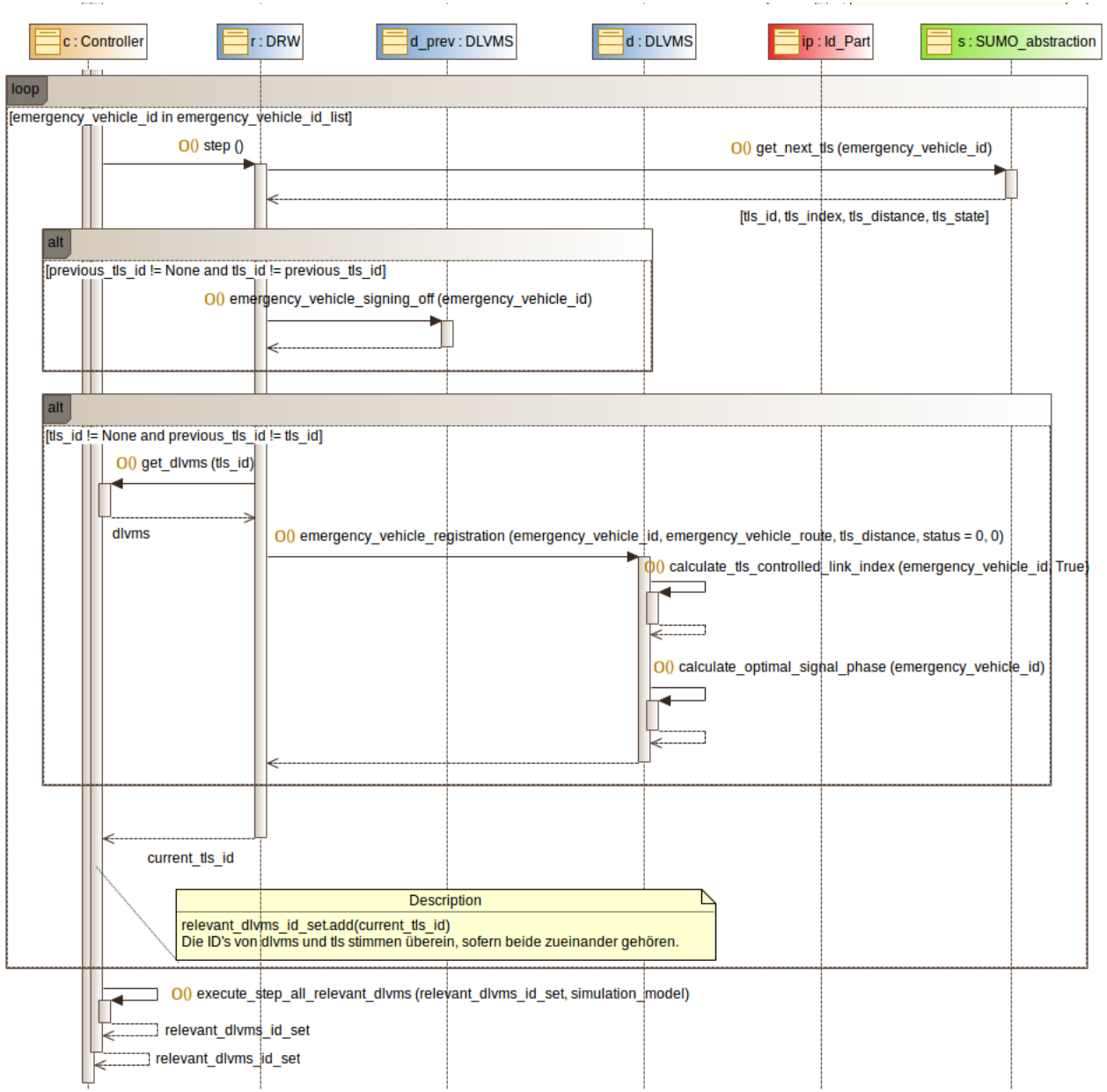
Setup der DLVMSe

Alarmfahrt DyReC

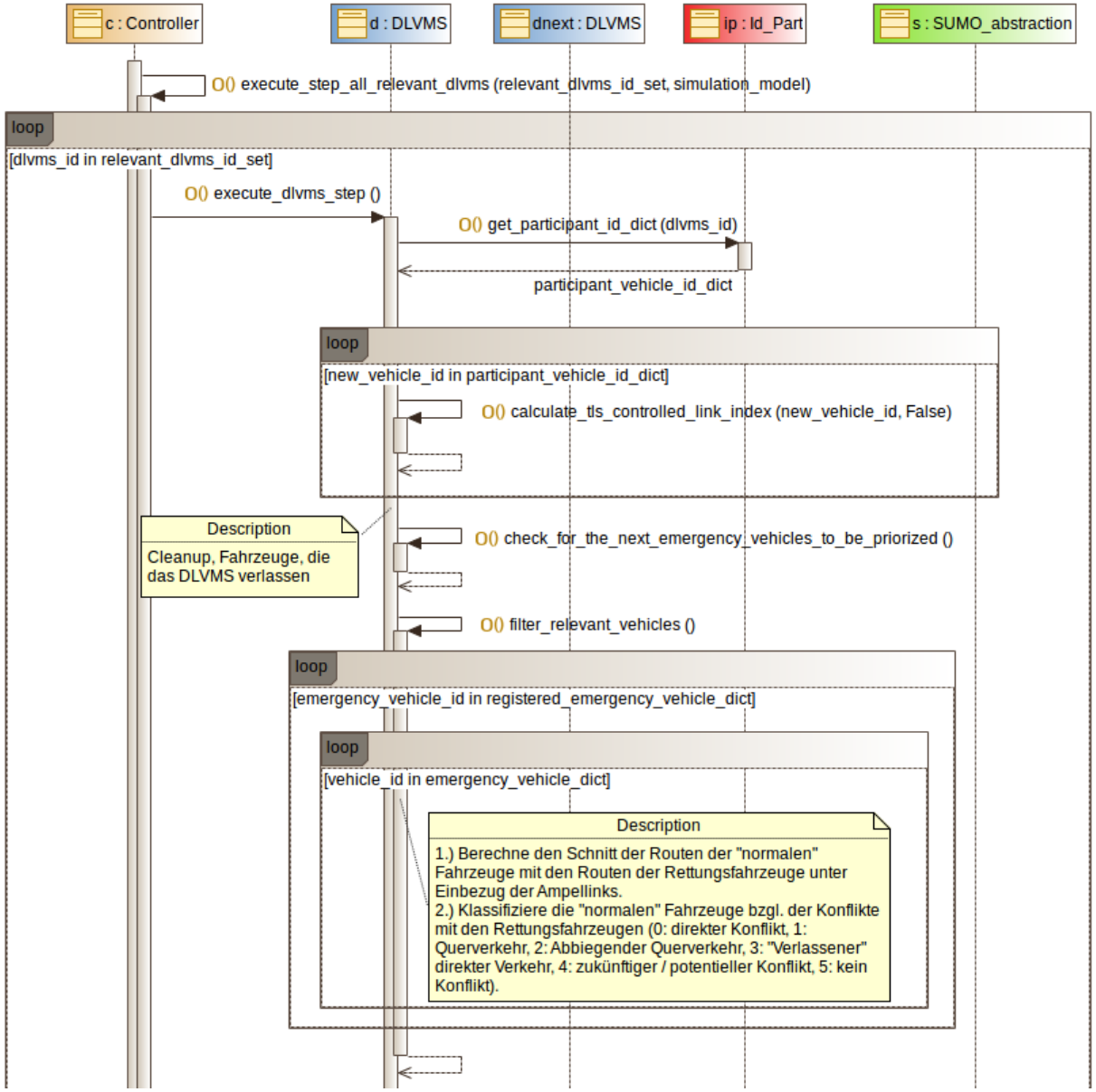


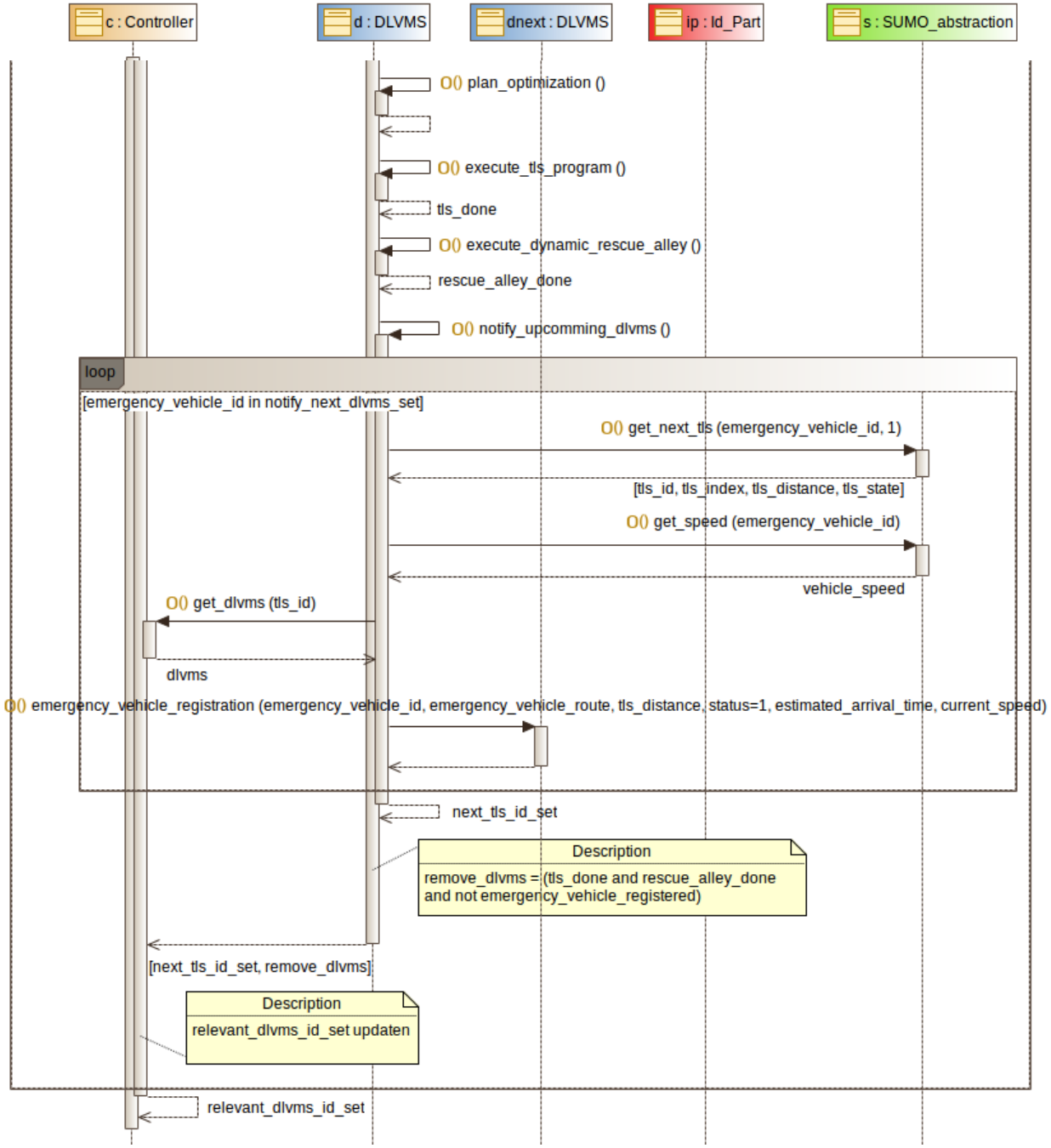
DyReC-Step im Controller



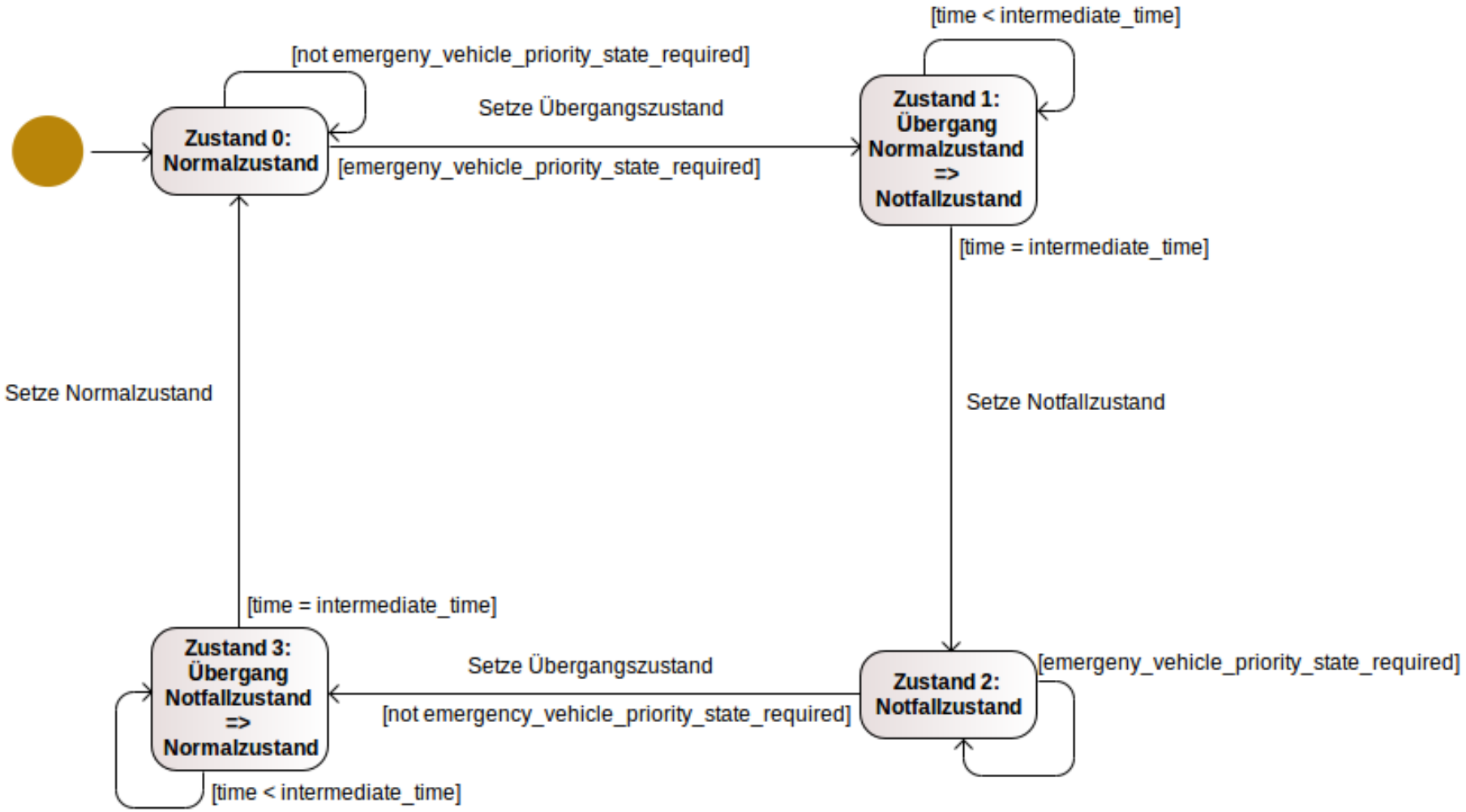


DyReC-Step im DLVMS

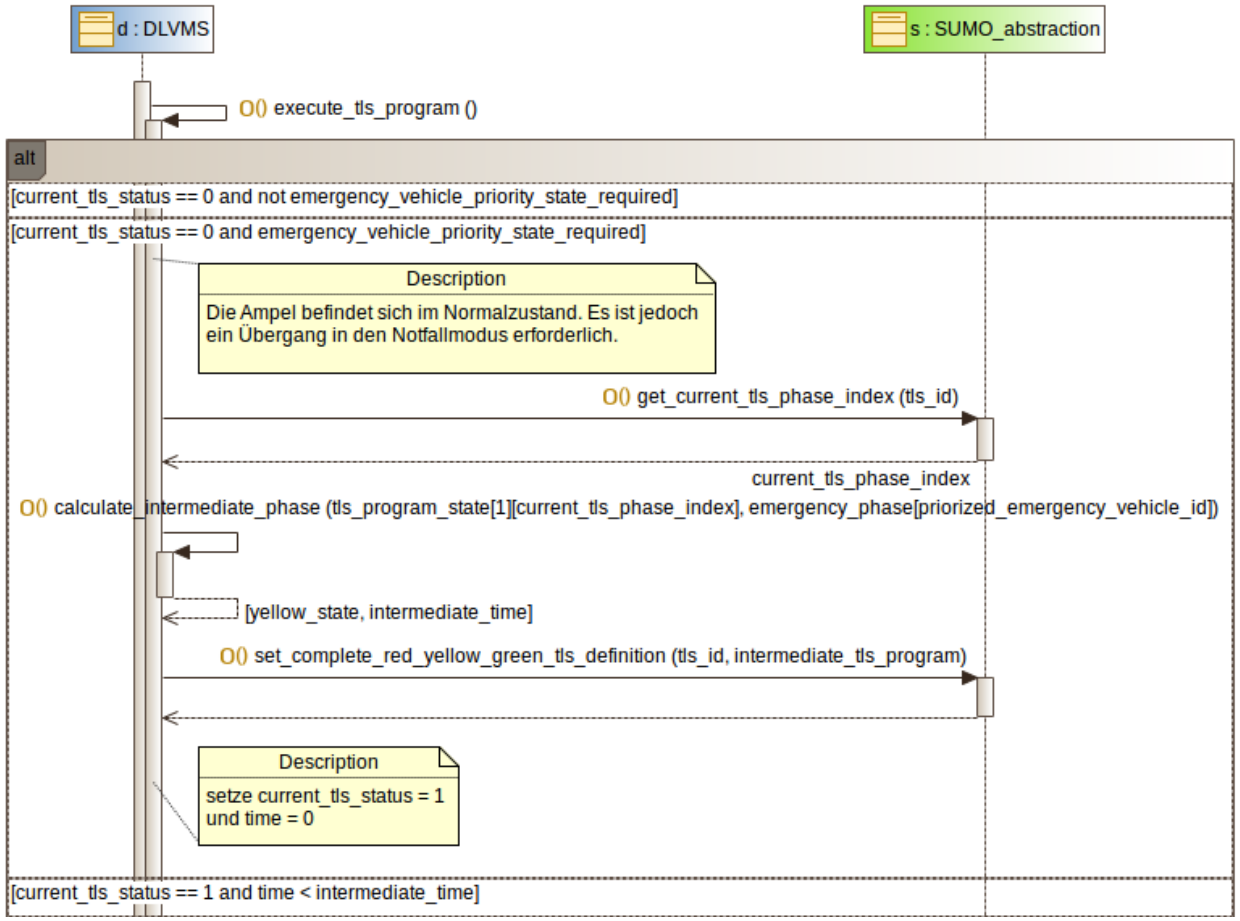


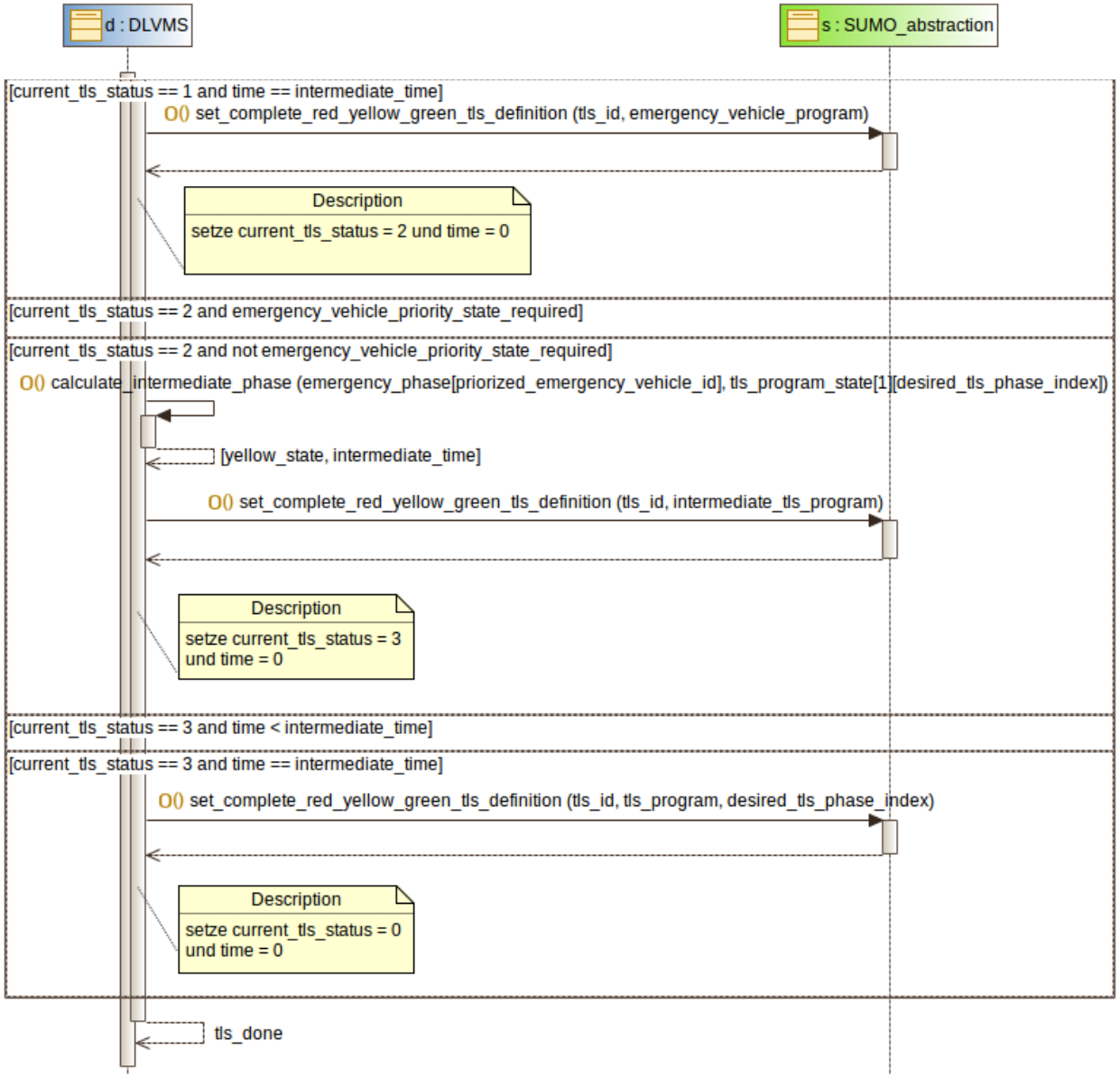


Automat zur LSA Anpassung

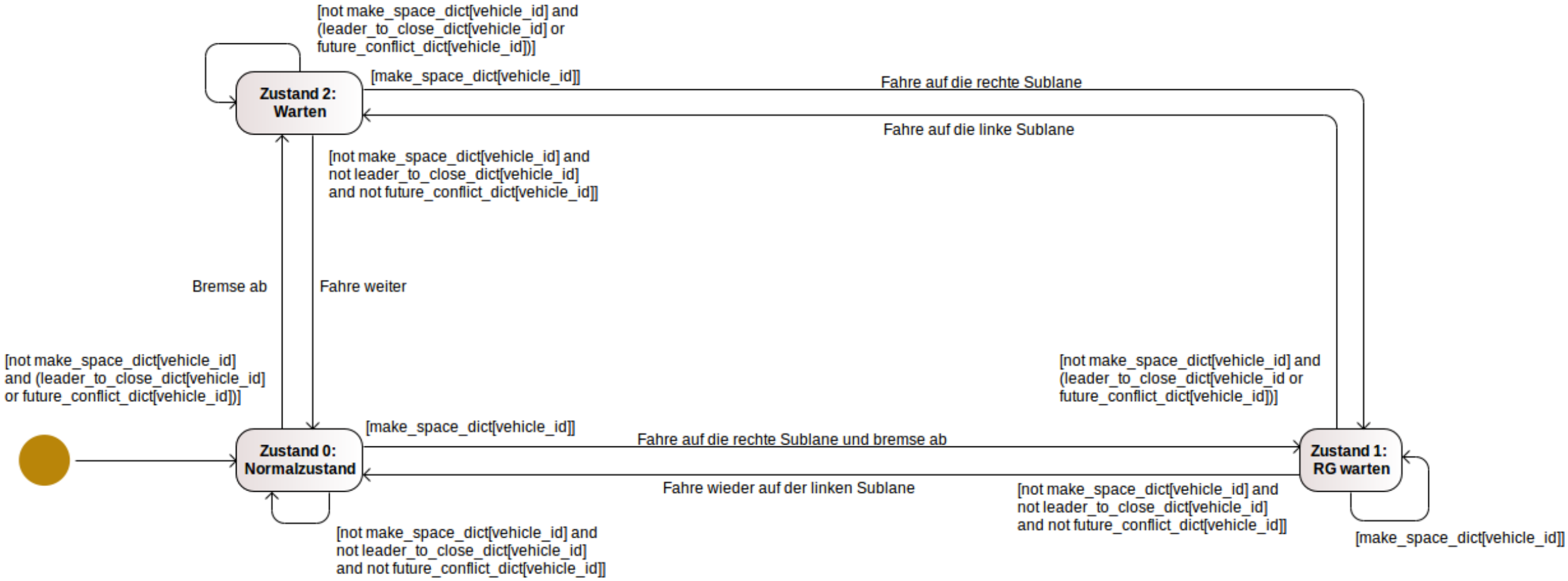


LSA Anpassung

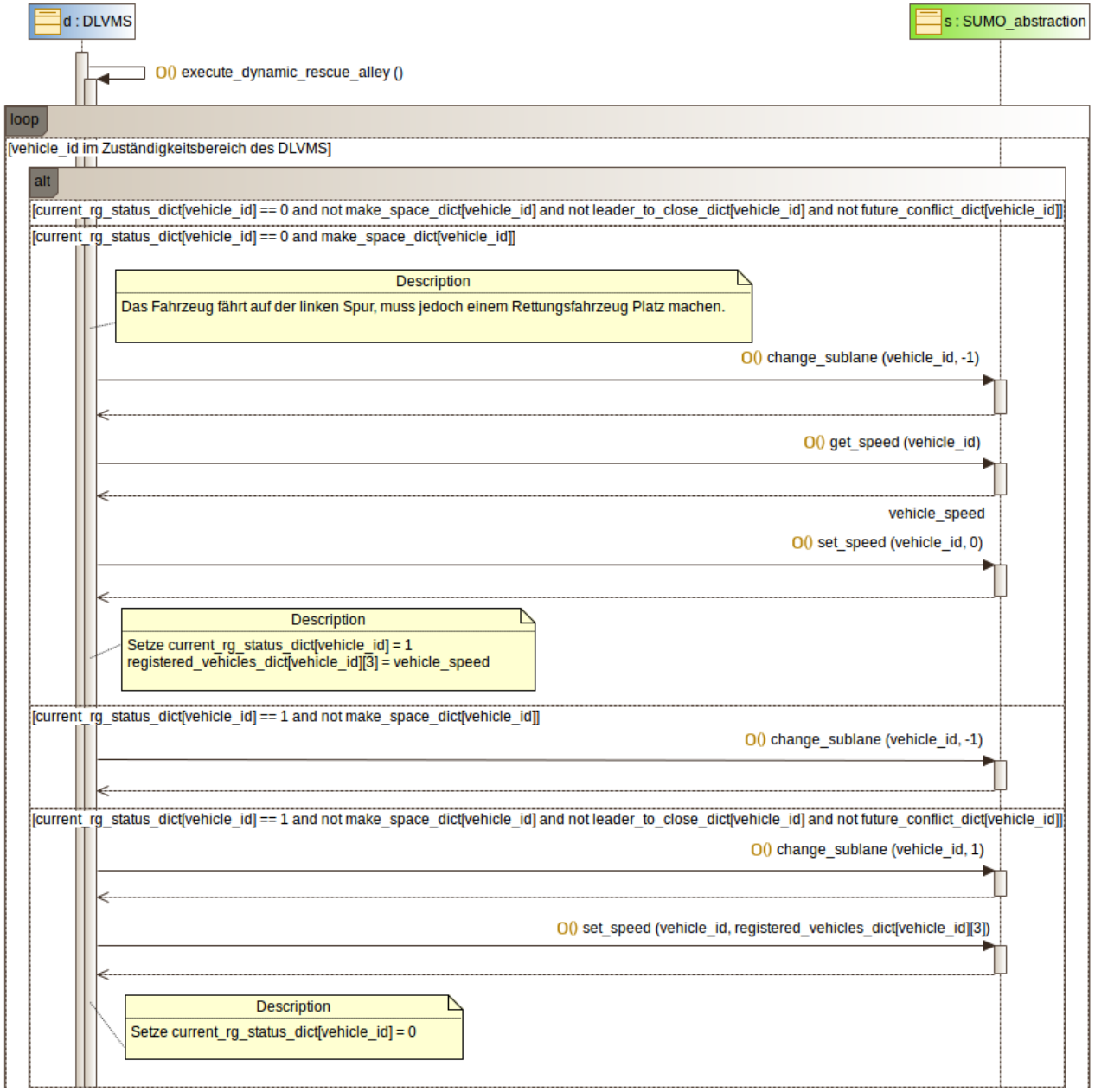


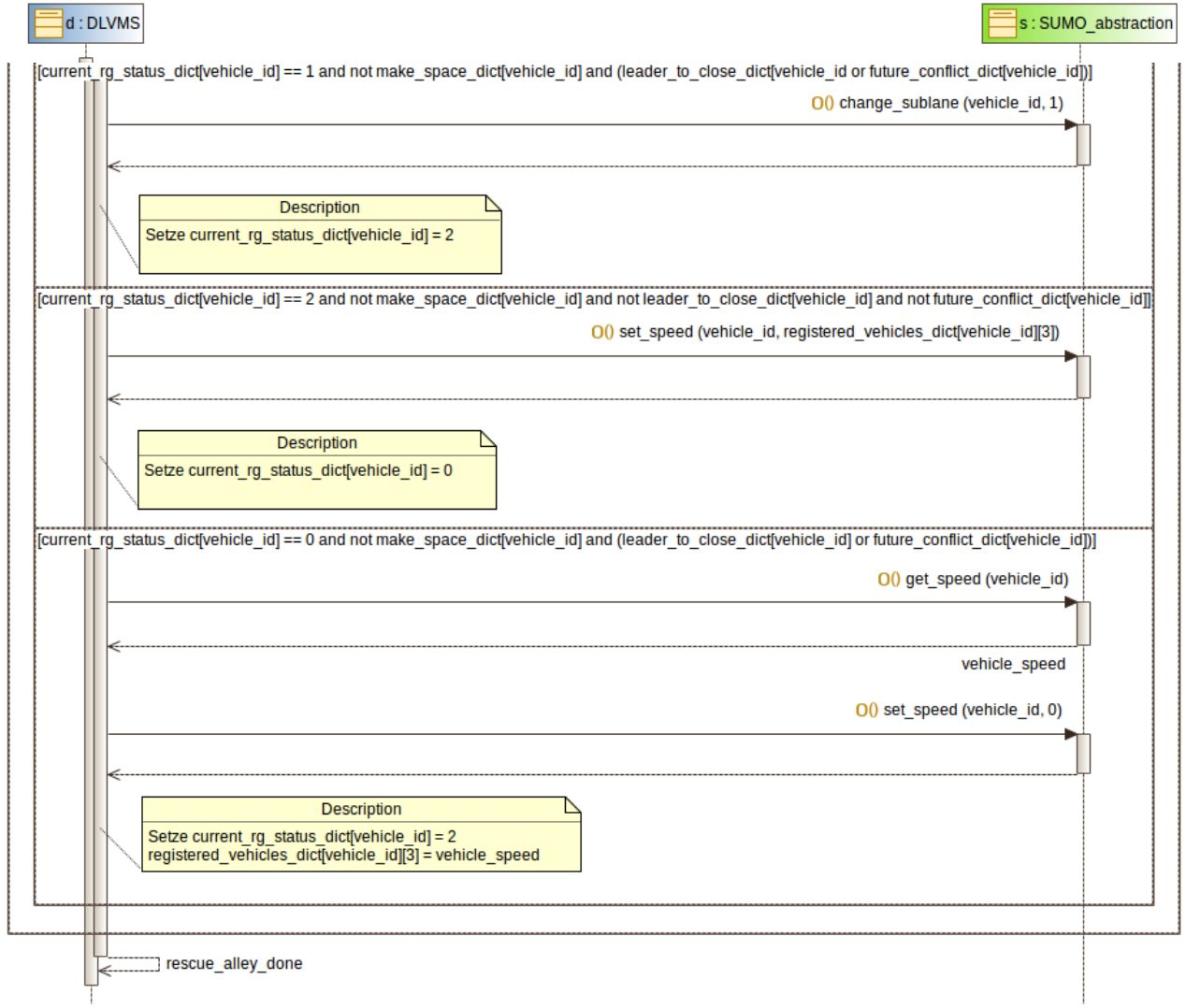


Automat zur Bildung der Rettungsgasse



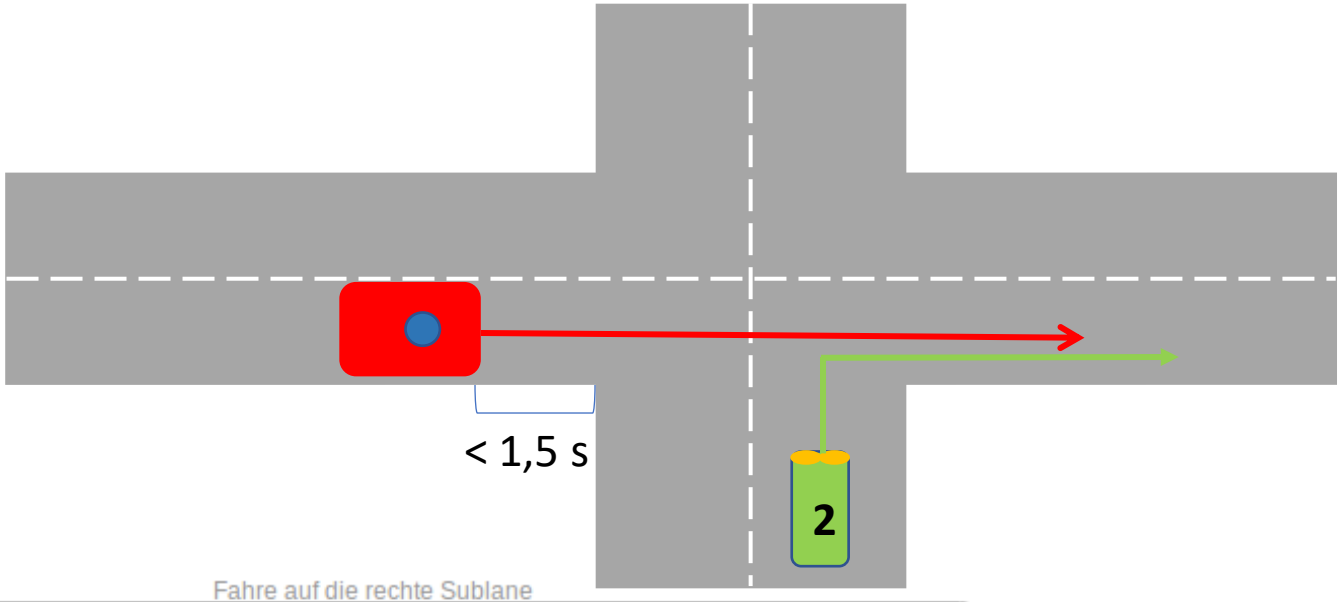
Bildung der Rettungsgasse

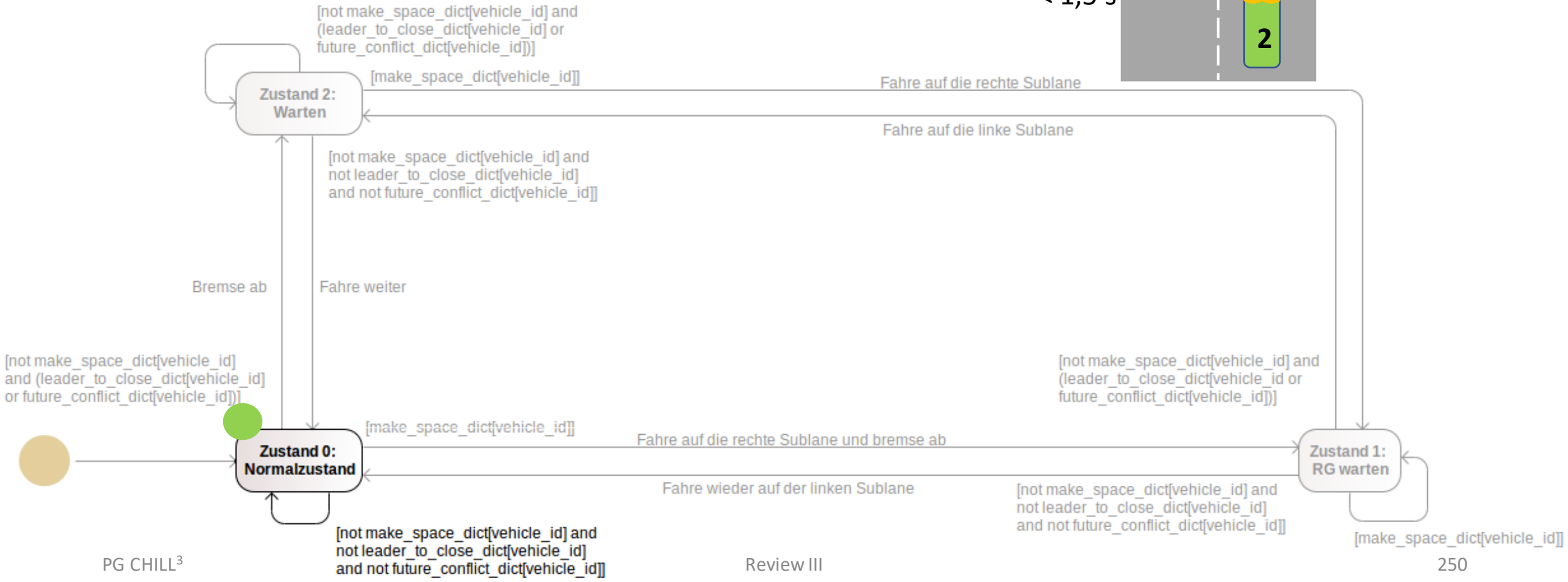




Zustandsautomat eines Fahrzeugs

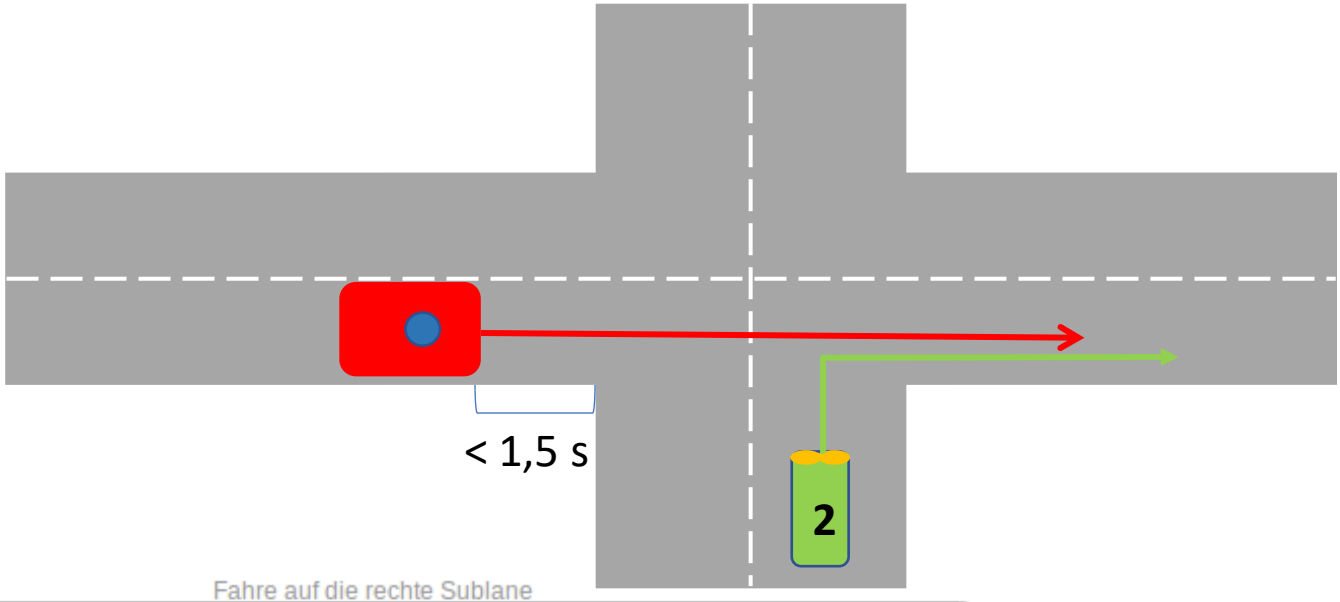
make_space_dict ={2: F}
leader_to_close_dict={2: F}
future_conflict_dict ={2: F}

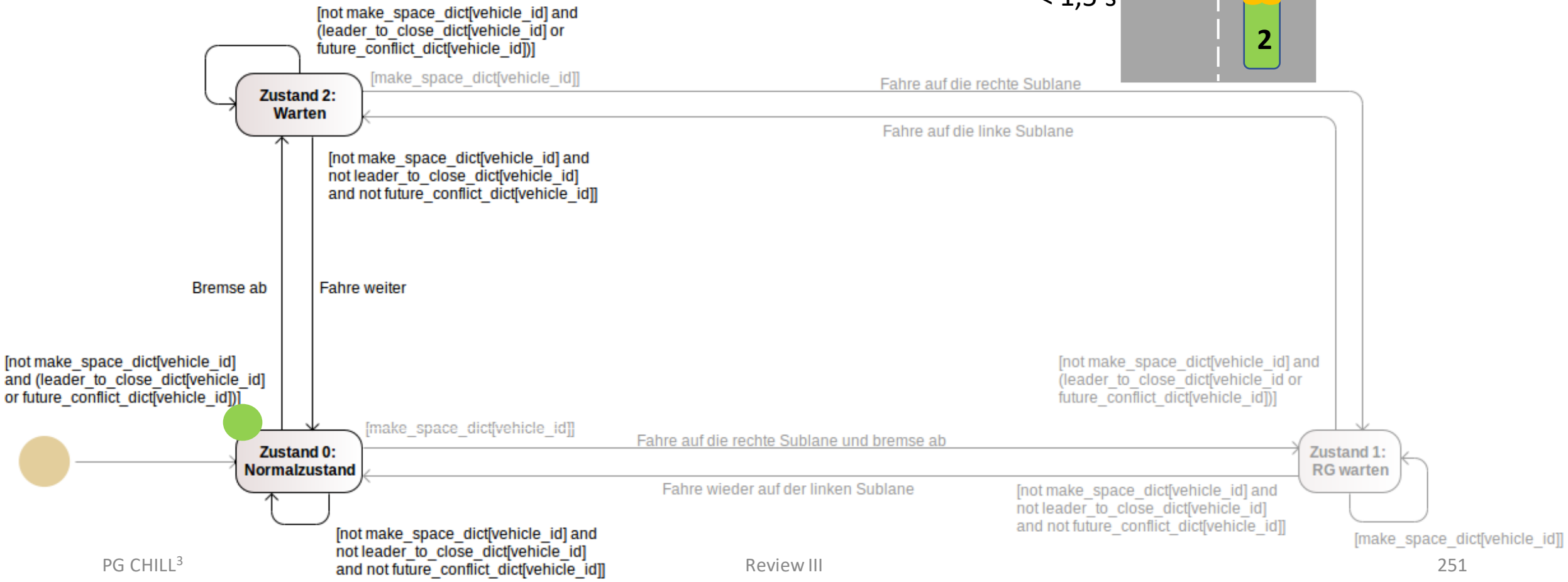




Zustandsautomat eines Fahrzeugs

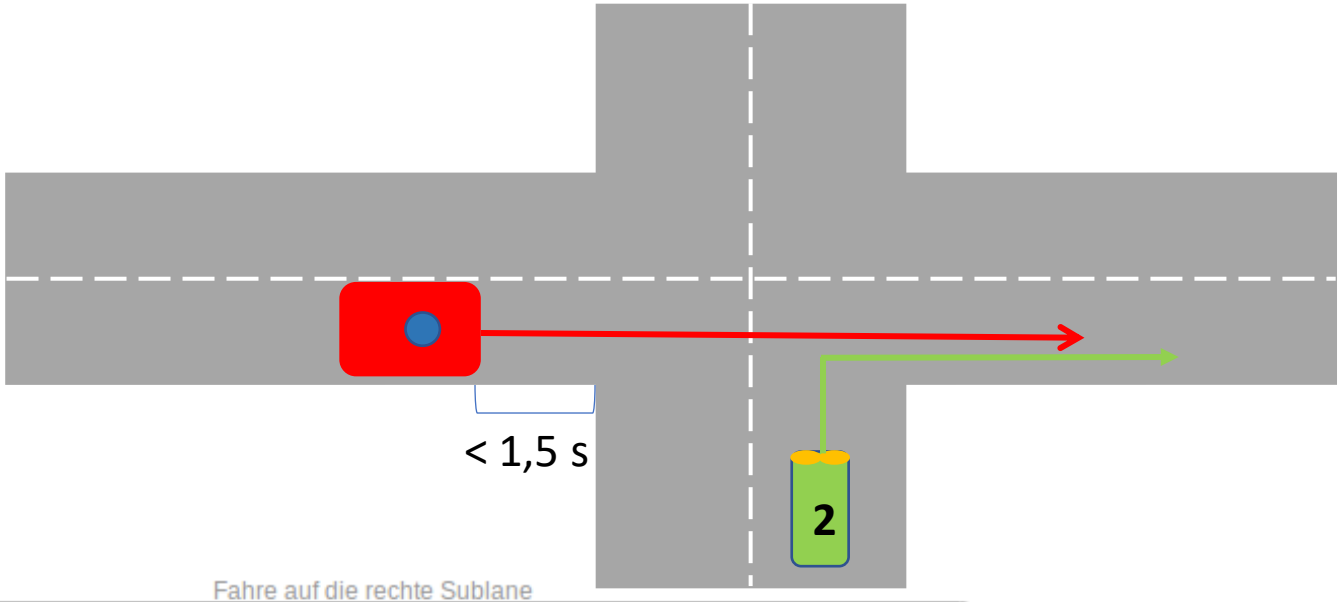
make_space_dict ={2: F}
leader_to_close_dict={2: F}
future_conflict_dict ={2: T}

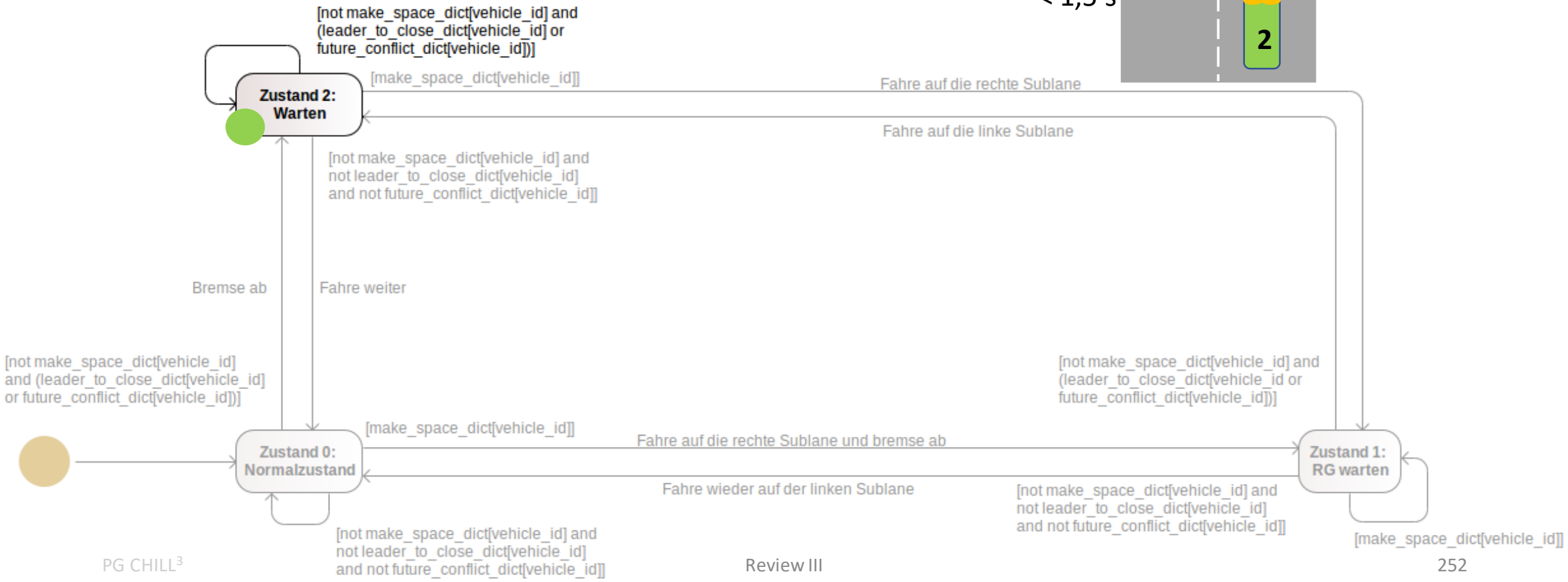




Zustandsautomat eines Fahrzeugs

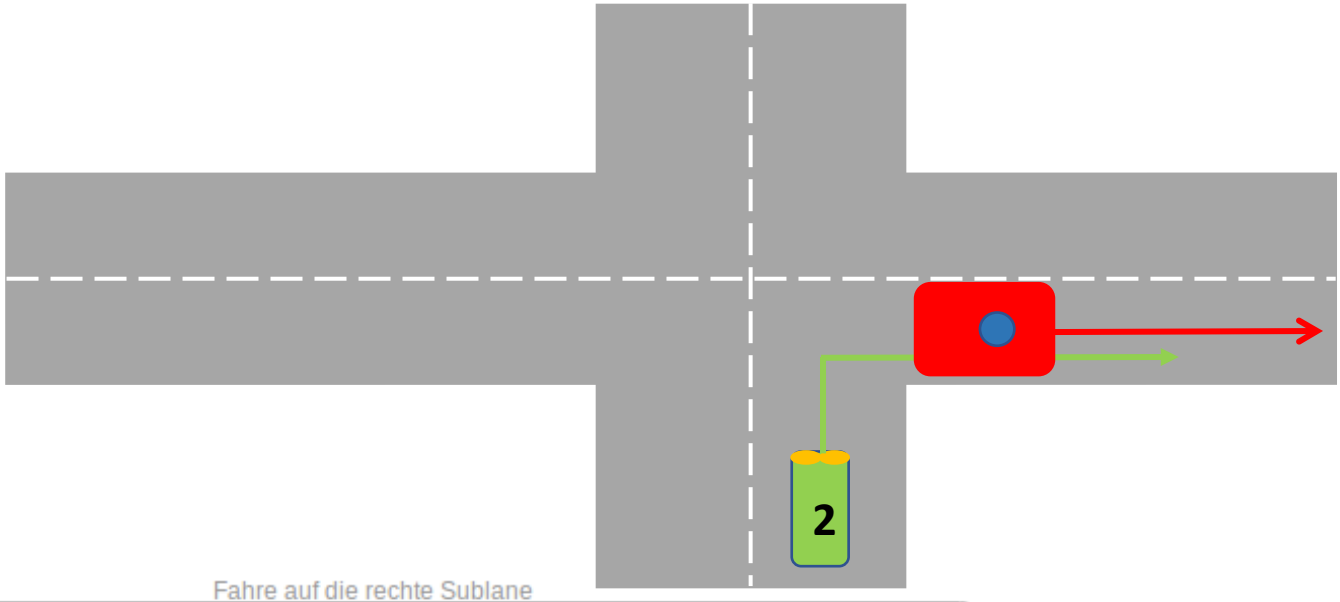
make_space_dict ={2: F}
leader_to_close_dict={2: F}
future_conflict_dict ={2: T}

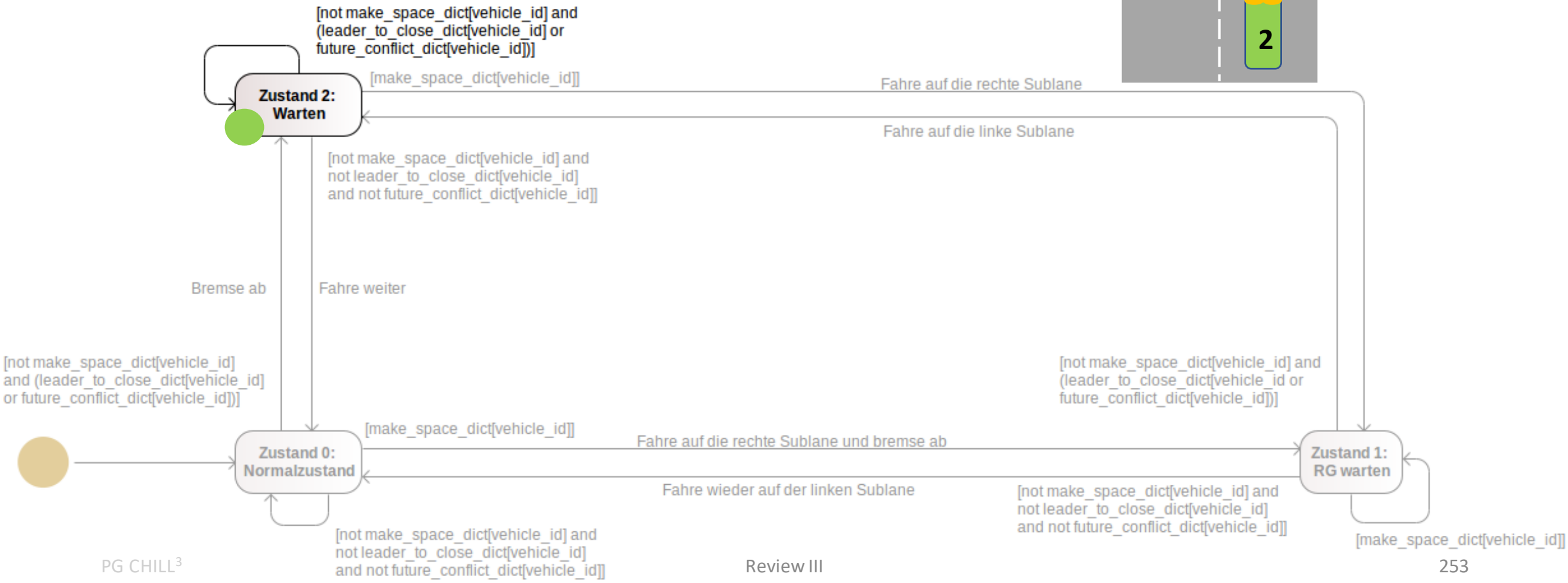




Zustandsautomat eines Fahrzeugs

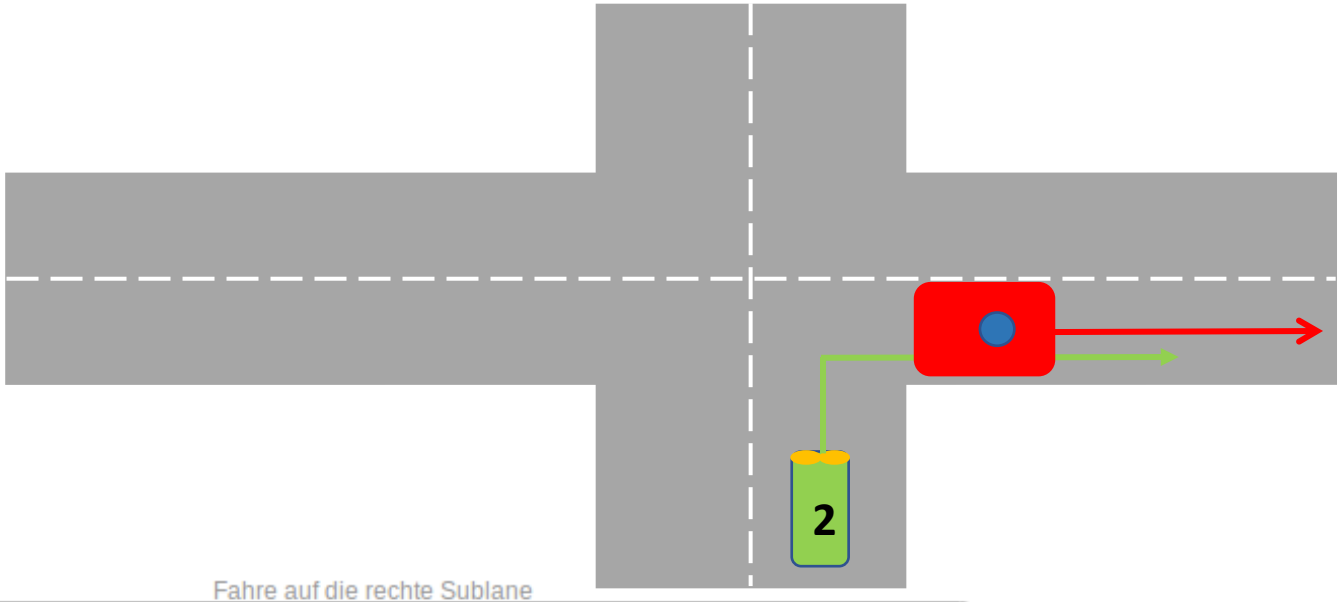
make_space_dict ={2: F}
leader_to_close_dict={2: F}
future_conflict_dict ={2: T}

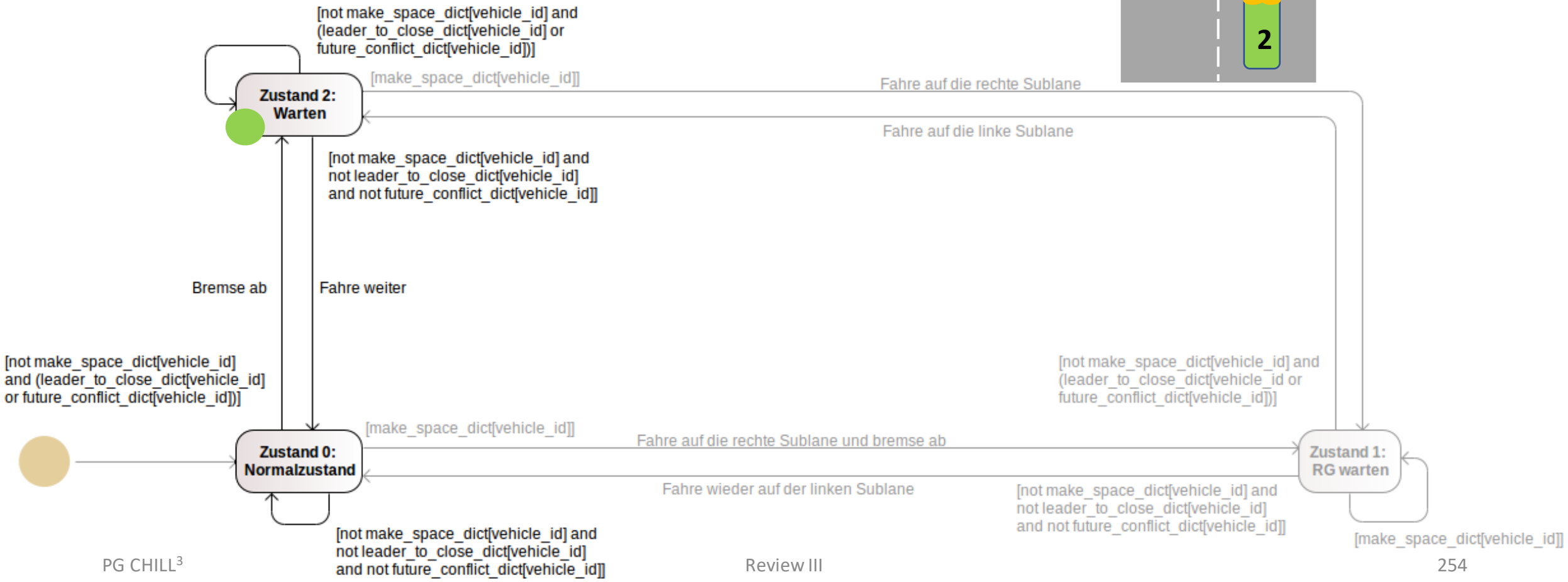




Zustandsautomat eines Fahrzeugs

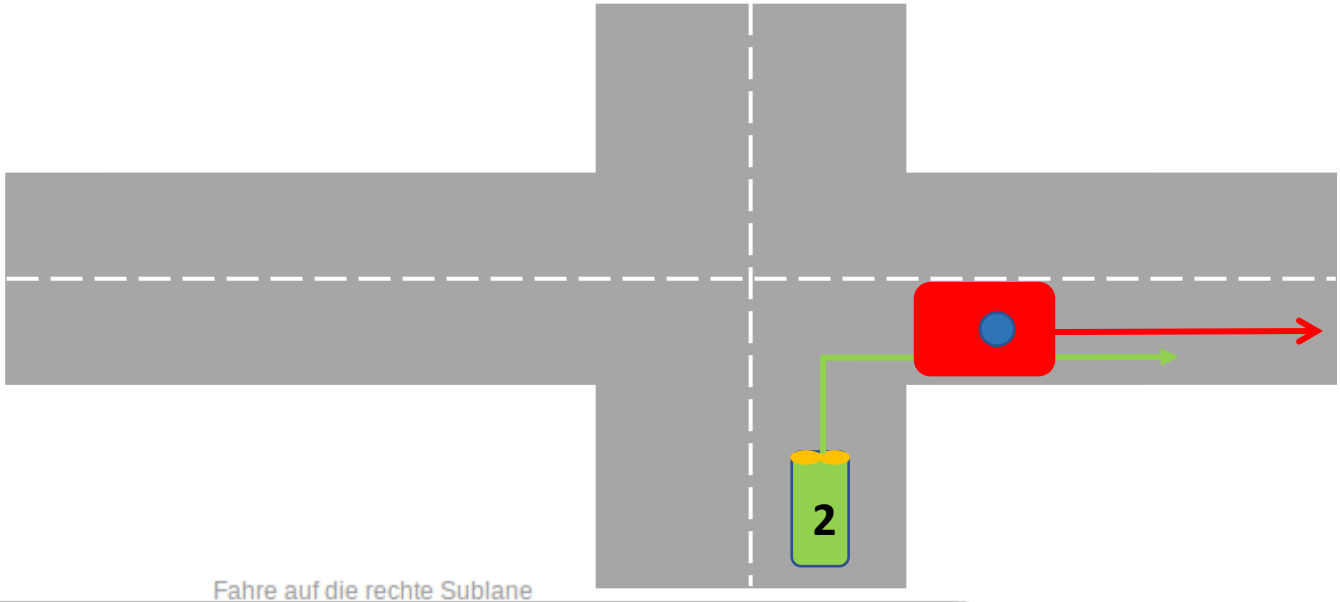
make_space_dict ={2: F}
leader_to_close_dict={2: F}
future_conflict_dict ={2: F}





Zustandsautomat eines Fahrzeugs

make_space_dict ={2: F}
leader_to_close_dict={2: F}
future_conflict_dict ={2: F}



[not make_space_dict[vehicle_id] and (leader_to_close_dict[vehicle_id] or future_conflict_dict[vehicle_id])]

[make_space_dict[vehicle_id]]

Zustand 2: Warten

Fahre auf die rechte Sublane

Fahre auf die linke Sublane

[not make_space_dict[vehicle_id] and not leader_to_close_dict[vehicle_id] and not future_conflict_dict[vehicle_id]]

Bremse ab

Fahre weiter

[not make_space_dict[vehicle_id] and (leader_to_close_dict[vehicle_id] or future_conflict_dict[vehicle_id])]

[not make_space_dict[vehicle_id] and (leader_to_close_dict[vehicle_id or future_conflict_dict[vehicle_id])]

[make_space_dict[vehicle_id]]

Zustand 0: Normalzustand

Fahre auf die rechte Sublane und bremse ab

Zustand 1: RG warten

Fahre wieder auf der linken Sublane

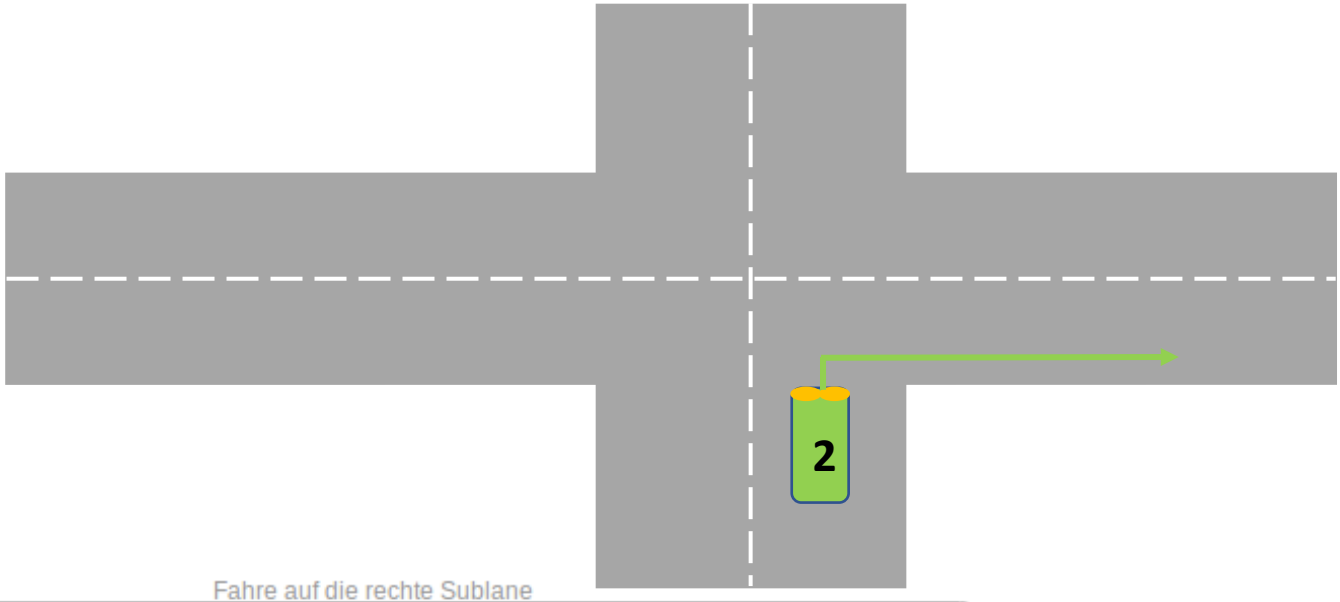
[not make_space_dict[vehicle_id] and not leader_to_close_dict[vehicle_id] and not future_conflict_dict[vehicle_id]]

[not make_space_dict[vehicle_id] and not leader_to_close_dict[vehicle_id] and not future_conflict_dict[vehicle_id]]

[make_space_dict[vehicle_id]]

Zustandsautomat eines Fahrzeugs

make_space_dict ={2: F}
leader_to_close_dict={2: F}
future_conflict_dict ={2: F}



[not make_space_dict[vehicle_id] and (leader_to_close_dict[vehicle_id] or future_conflict_dict[vehicle_id])]

Zustand 2: Warten

[make_space_dict[vehicle_id]]

Fahre auf die rechte Sublane

Fahre auf die linke Sublane

[not make_space_dict[vehicle_id] and not leader_to_close_dict[vehicle_id] and not future_conflict_dict[vehicle_id]]

Bremse ab

Fahre weiter

[not make_space_dict[vehicle_id] and (leader_to_close_dict[vehicle_id] or future_conflict_dict[vehicle_id])]

[not make_space_dict[vehicle_id] and (leader_to_close_dict[vehicle_id or future_conflict_dict[vehicle_id])]

Zustand 0: Normalzustand

[make_space_dict[vehicle_id]]

Fahre auf die rechte Sublane und bremse ab

Zustand 1: RG warten

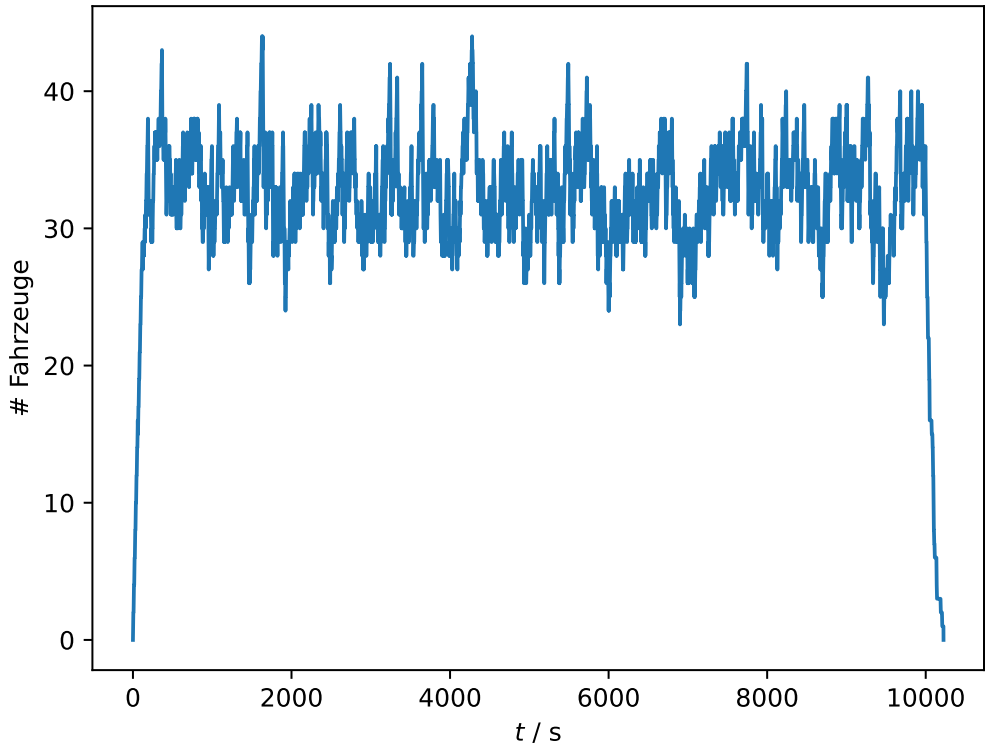
Fahre wieder auf der linken Sublane

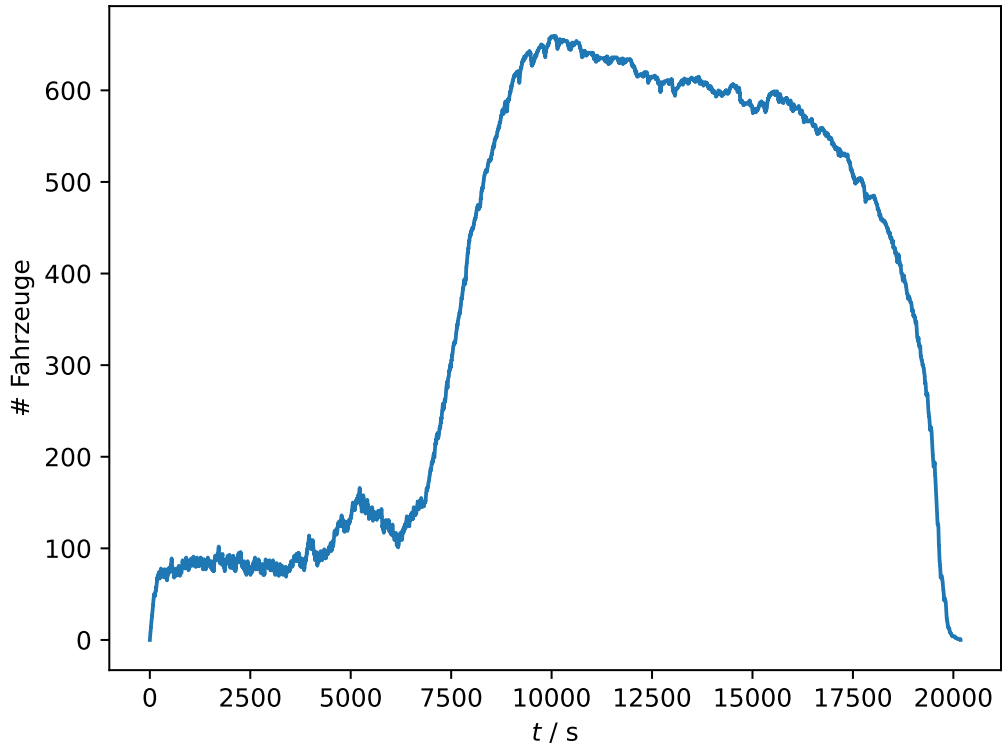
[not make_space_dict[vehicle_id] and not leader_to_close_dict[vehicle_id] and not future_conflict_dict[vehicle_id]]

[make_space_dict[vehicle_id]]

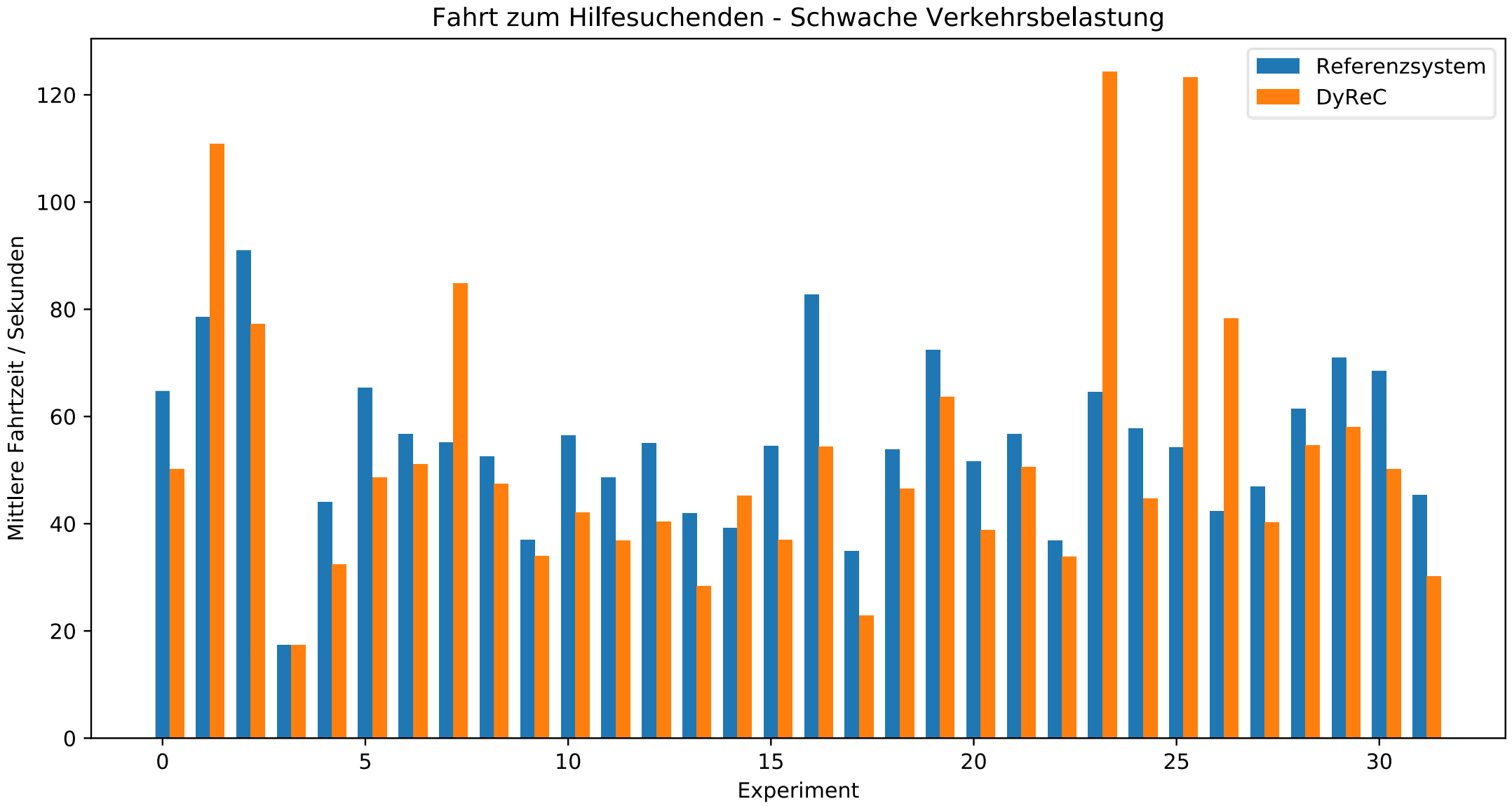
[not make_space_dict[vehicle_id] and not leader_to_close_dict[vehicle_id] and not future_conflict_dict[vehicle_id]]

Equilibrierung

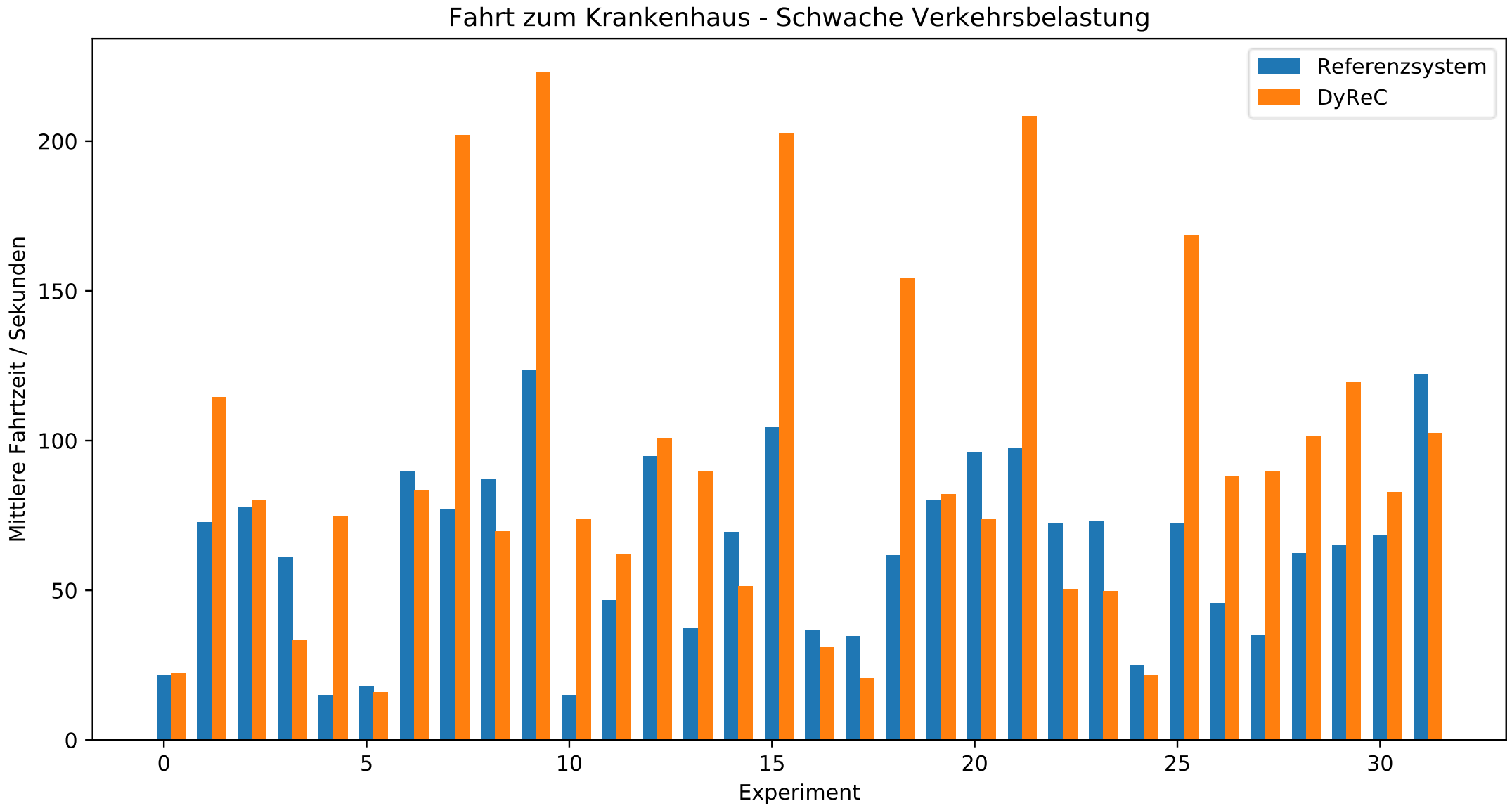




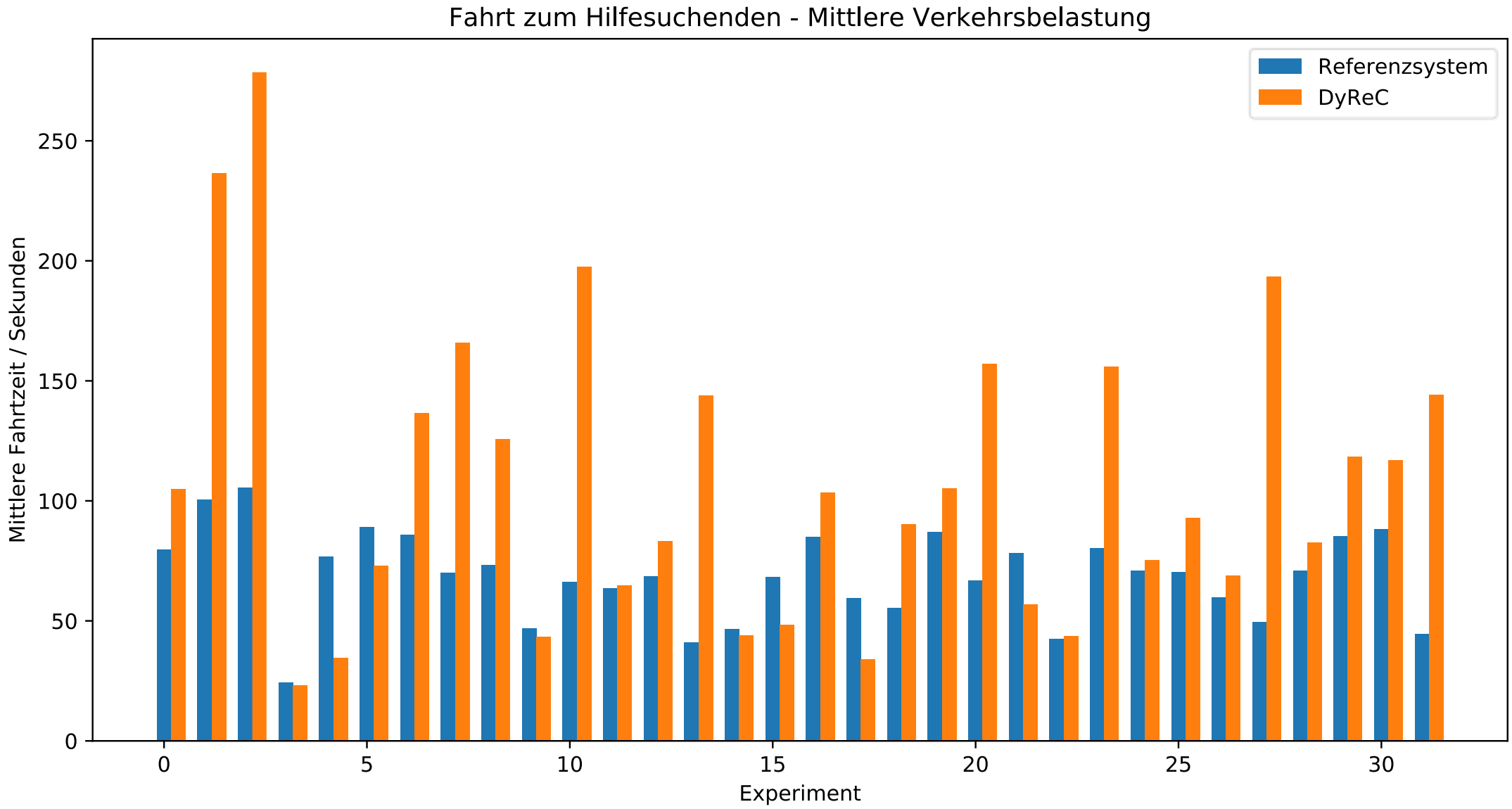
Ergebnisse schwache Verkehrsbelastung



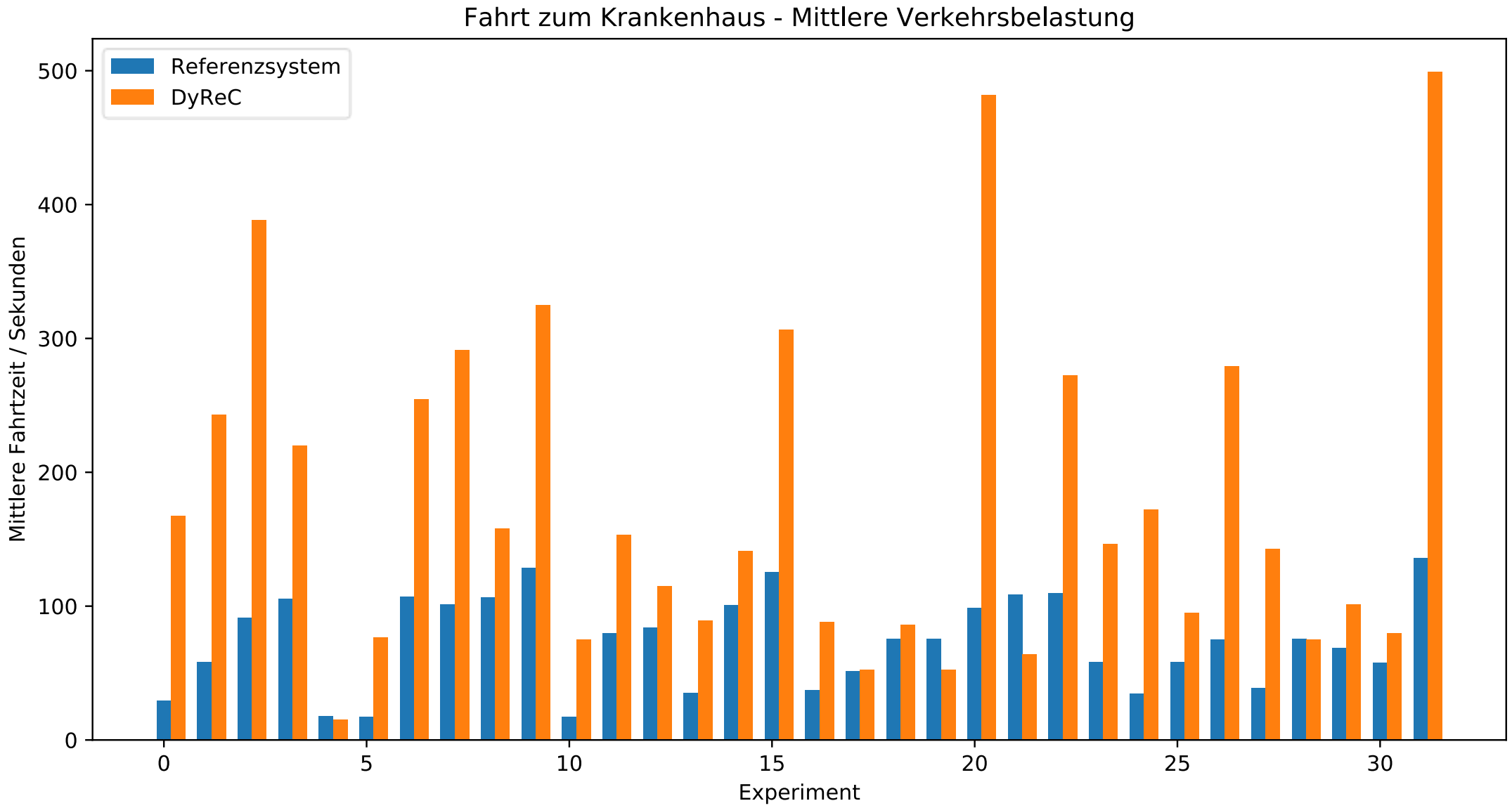
Ergebnisse schwache Verkehrsbelastung



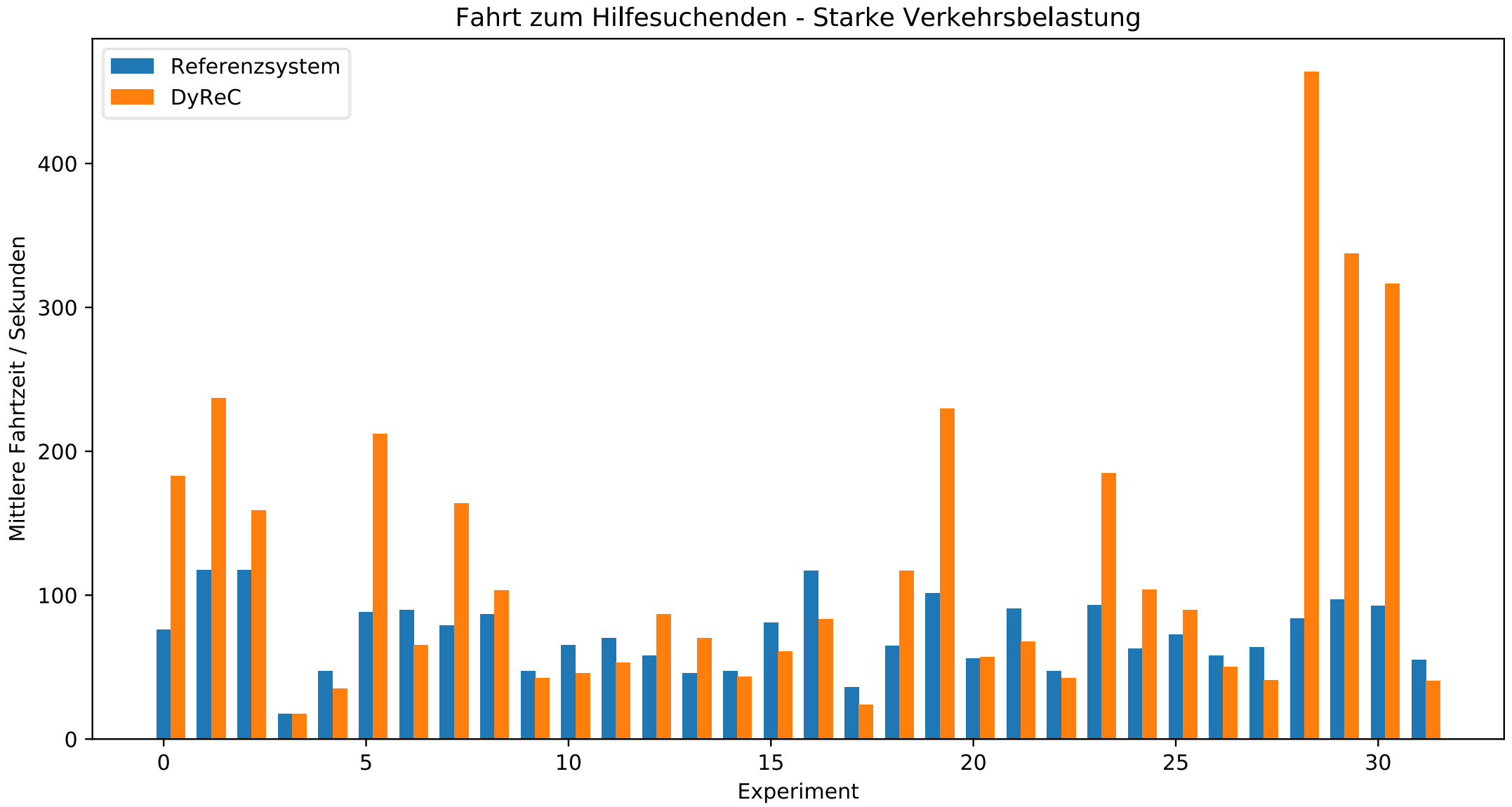
Ergebnisse mittlere Verkehrsbelastung



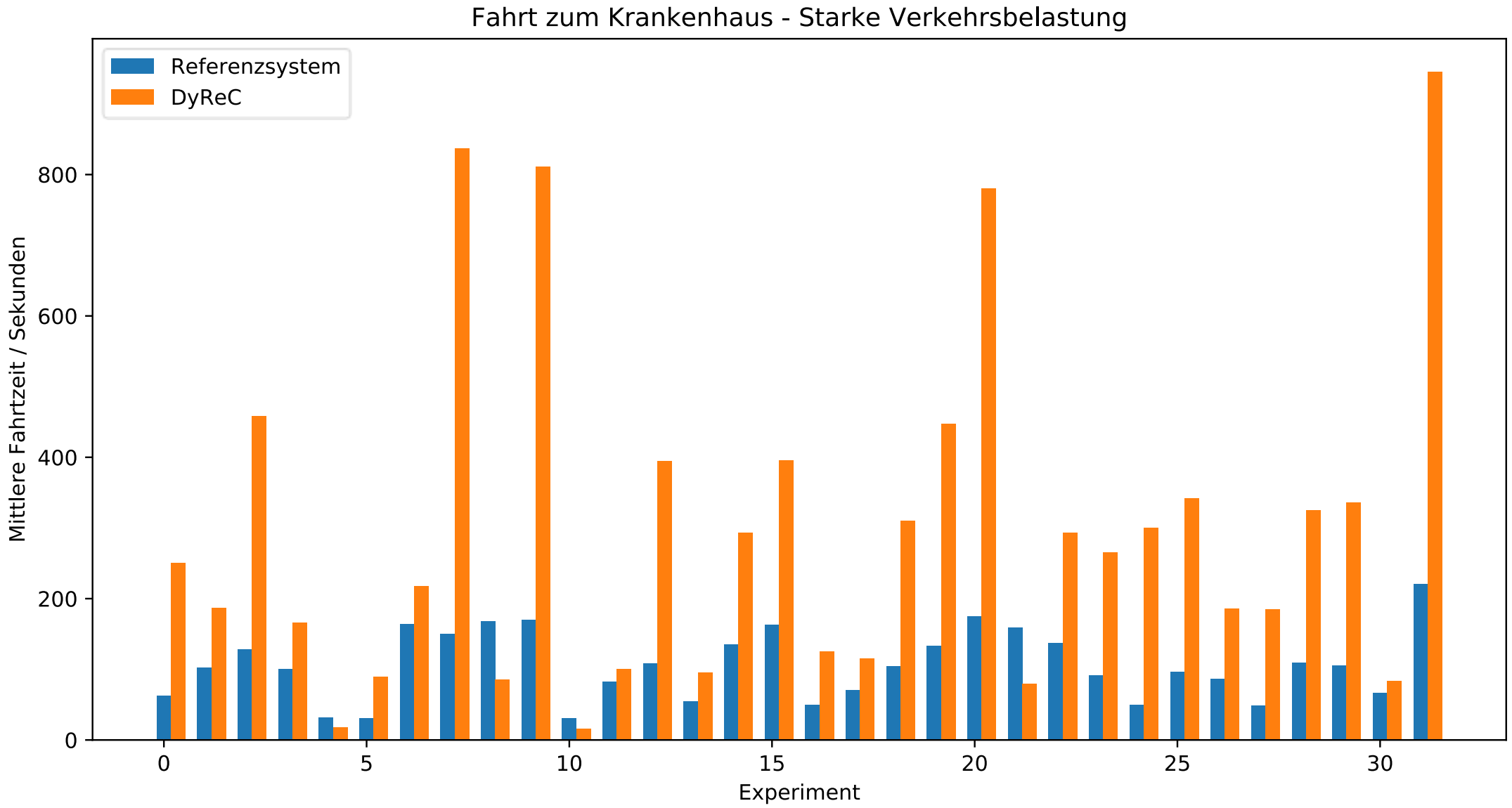
Ergebnisse mittlere Verkehrsbelastung



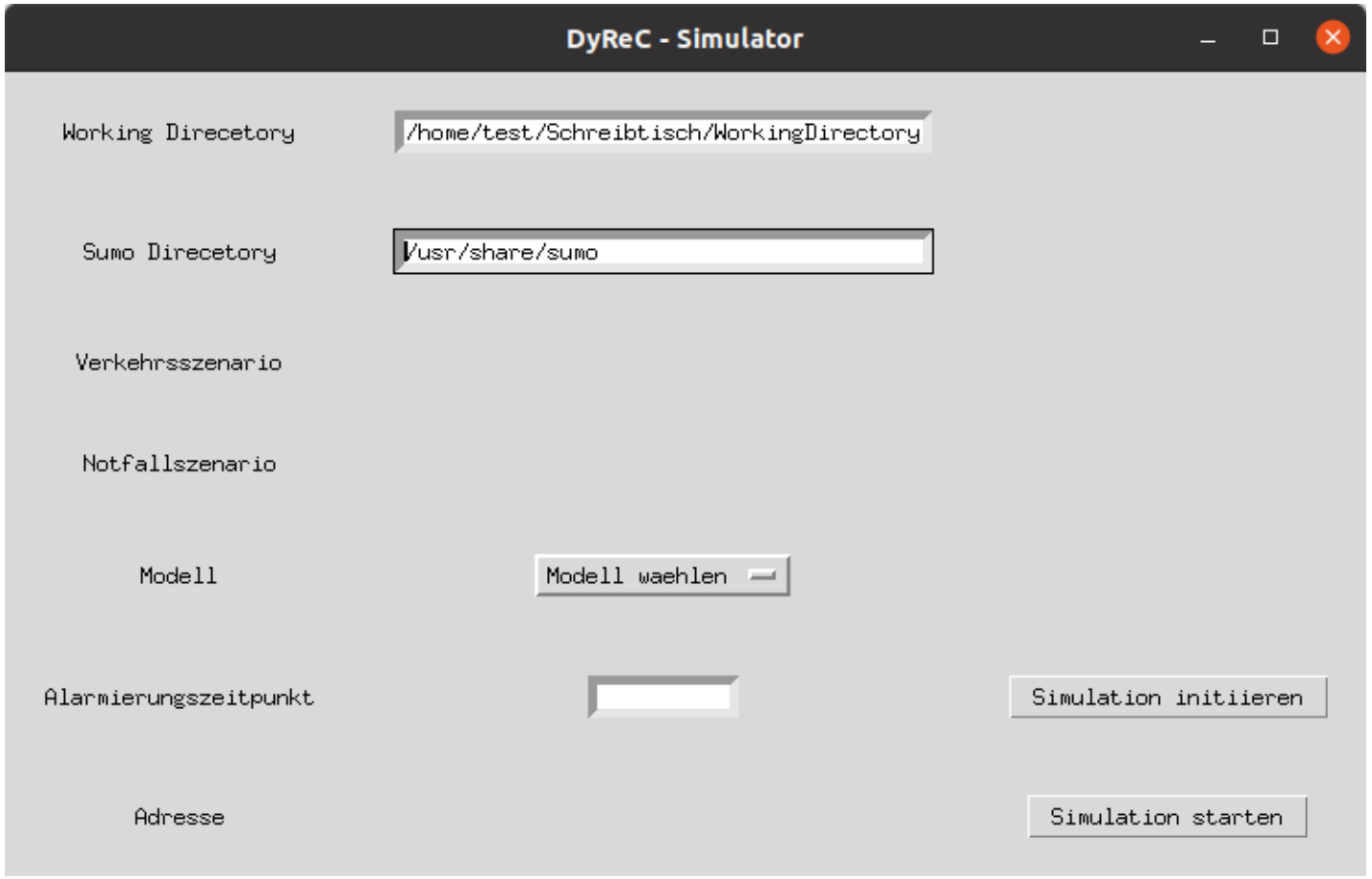
Ergebnisse starke Verkehrsbelastung



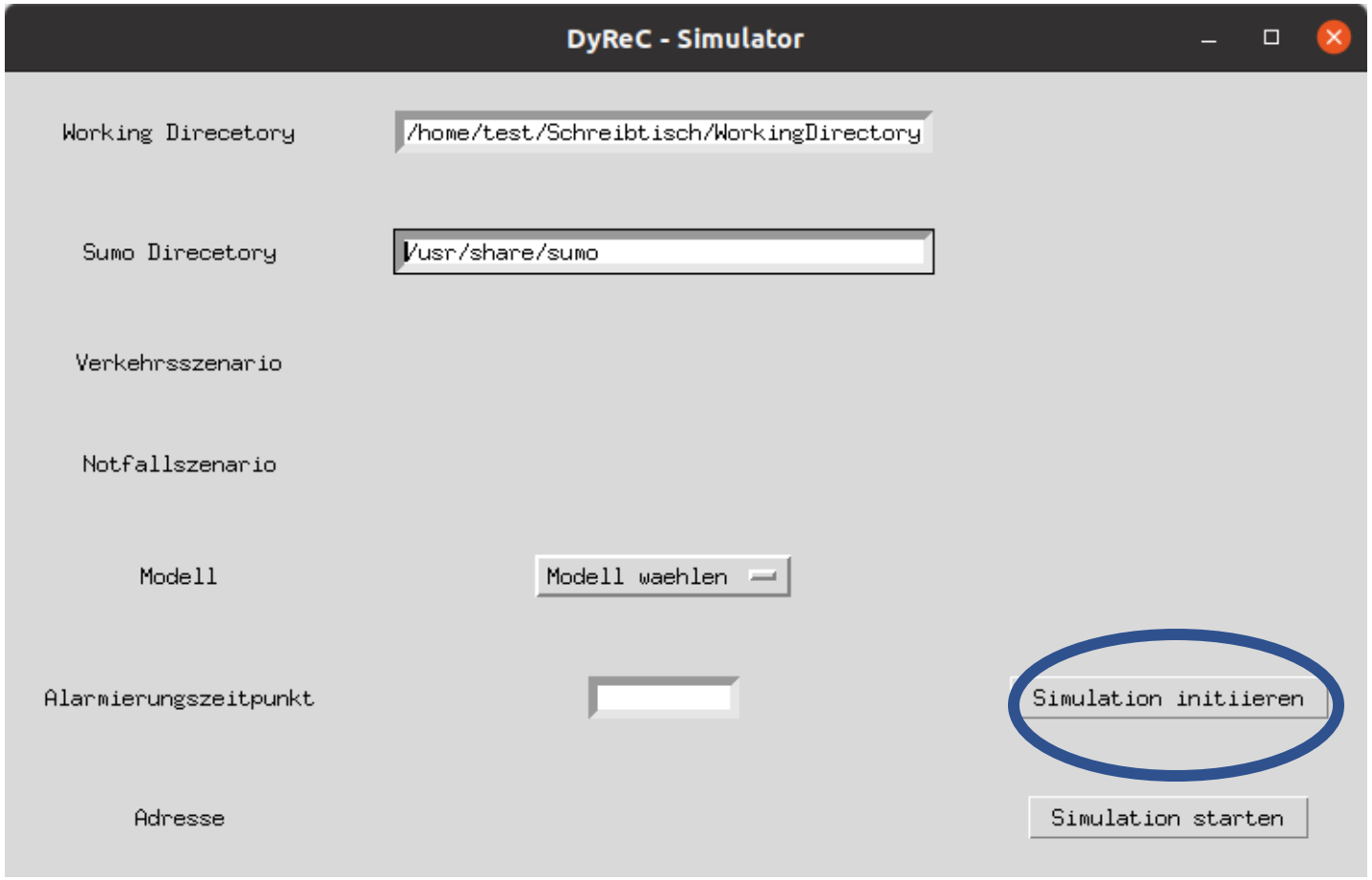
Ergebnisse starke Verkehrsbelastung



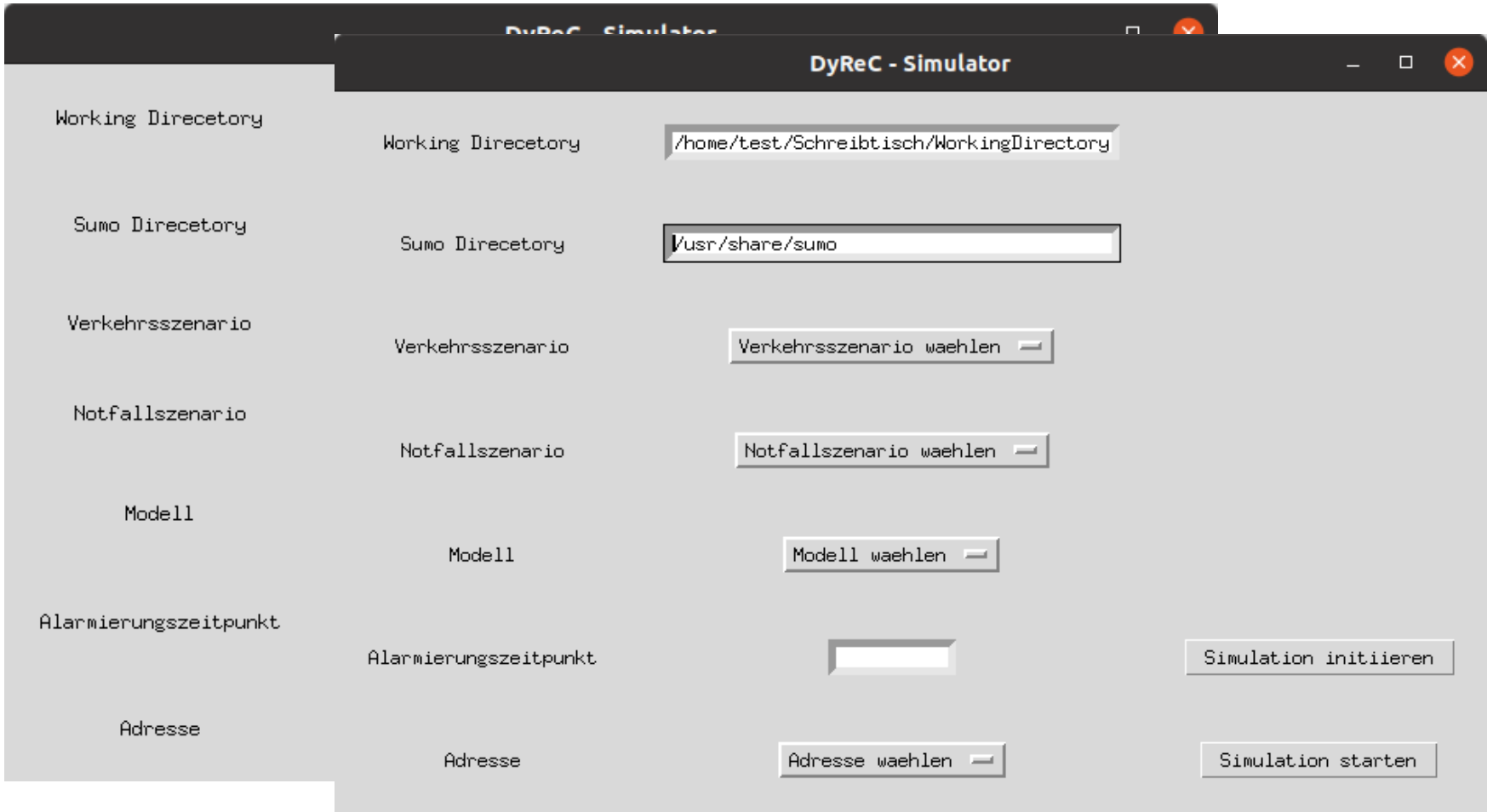
GUI


DyReC - Simulator
Working Direcetory
/home/test/Schreibtisch/WorkingDirectory
Sumo Direcetory
/usr/share/sumo
Verkehrsszenario
Notfallszenario
Modell
Modell waehlen
Alarmierungszeitpunkt
Simulation initiieren
Adresse
Simulation starten

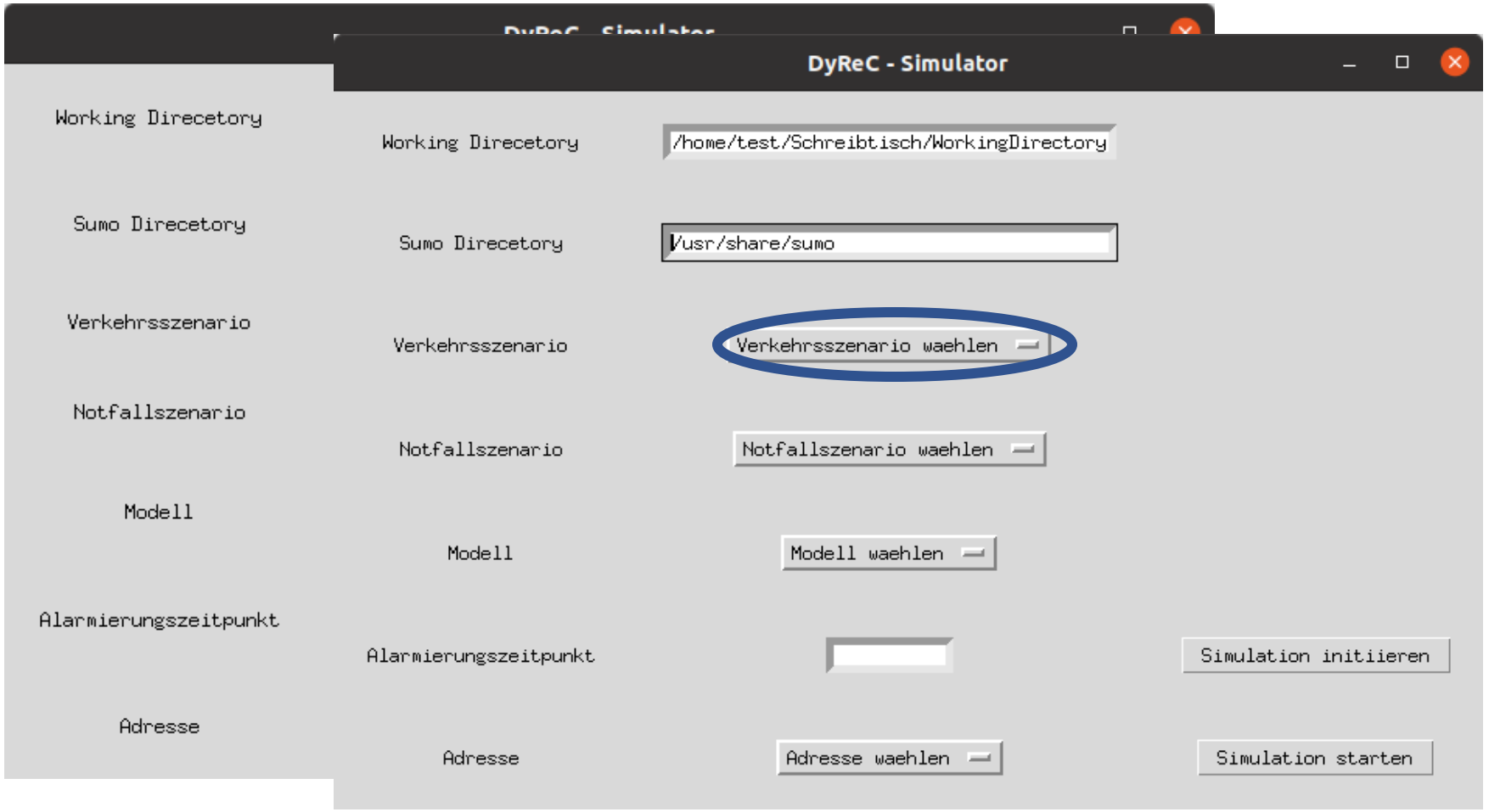
GUI


DyReC - Simulator
Working Direcetory
/home/test/Schreibtisch/WorkingDirectory
Sumo Direcetory
/usr/share/sumo
Verkehrsszenario
Notfallszenario
Modell
Modell waehlen
Alarmierungszeitpunkt
Simulation initiieren
Adresse
Simulation starten

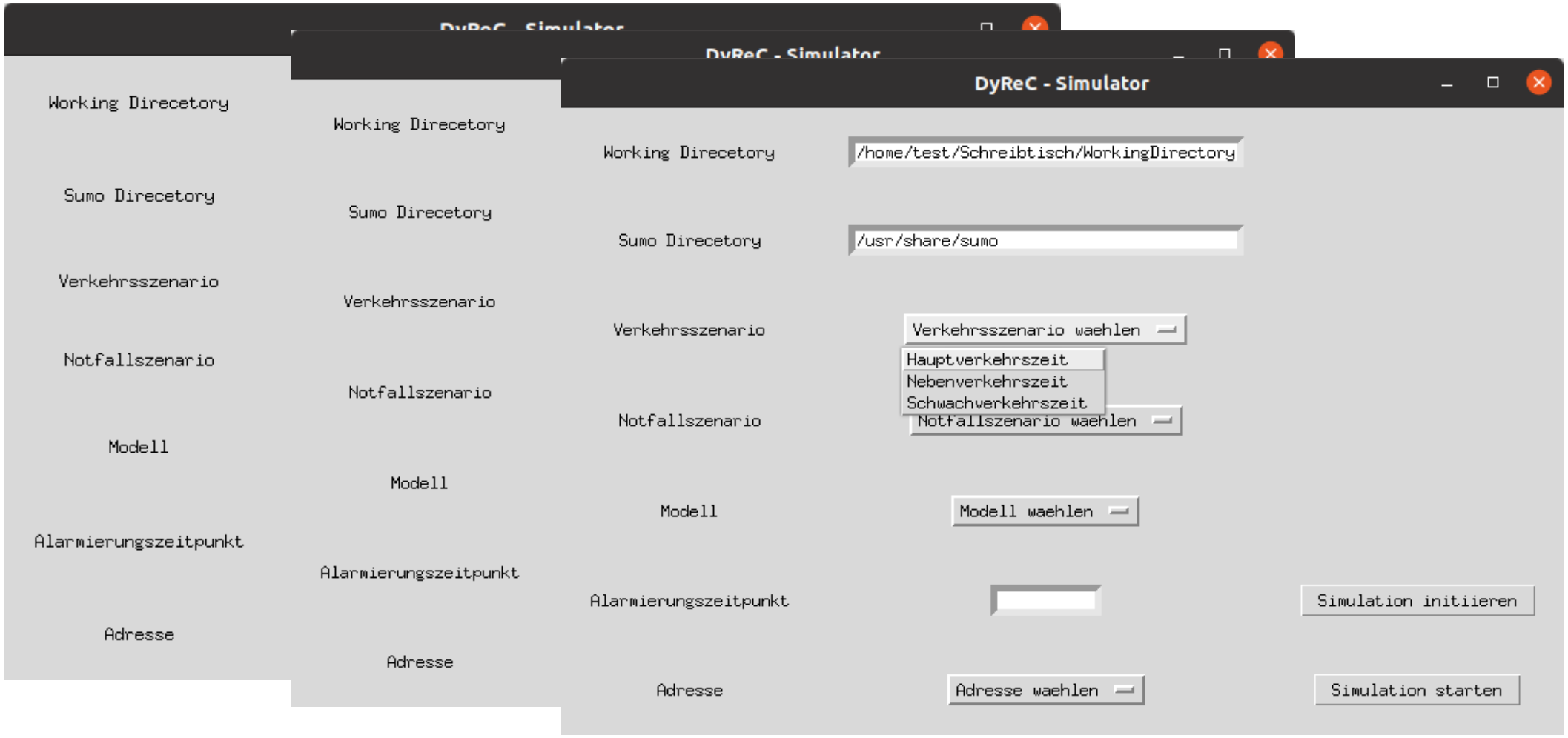
GUI


DyReC - Simulator
Working Direcetory
/home/test/Schreibtisch/WorkingDirectory
Sumo Direcetory
/usr/share/sumo
Verkehrsszenario
Verkehrsszenario waehlen
Notfallszenario
Notfallszenario waehlen
Modell
Modell waehlen
Alarmierungszeitpunkt
Simulation initiieren
Adresse
Adresse waehlen
Simulation starten

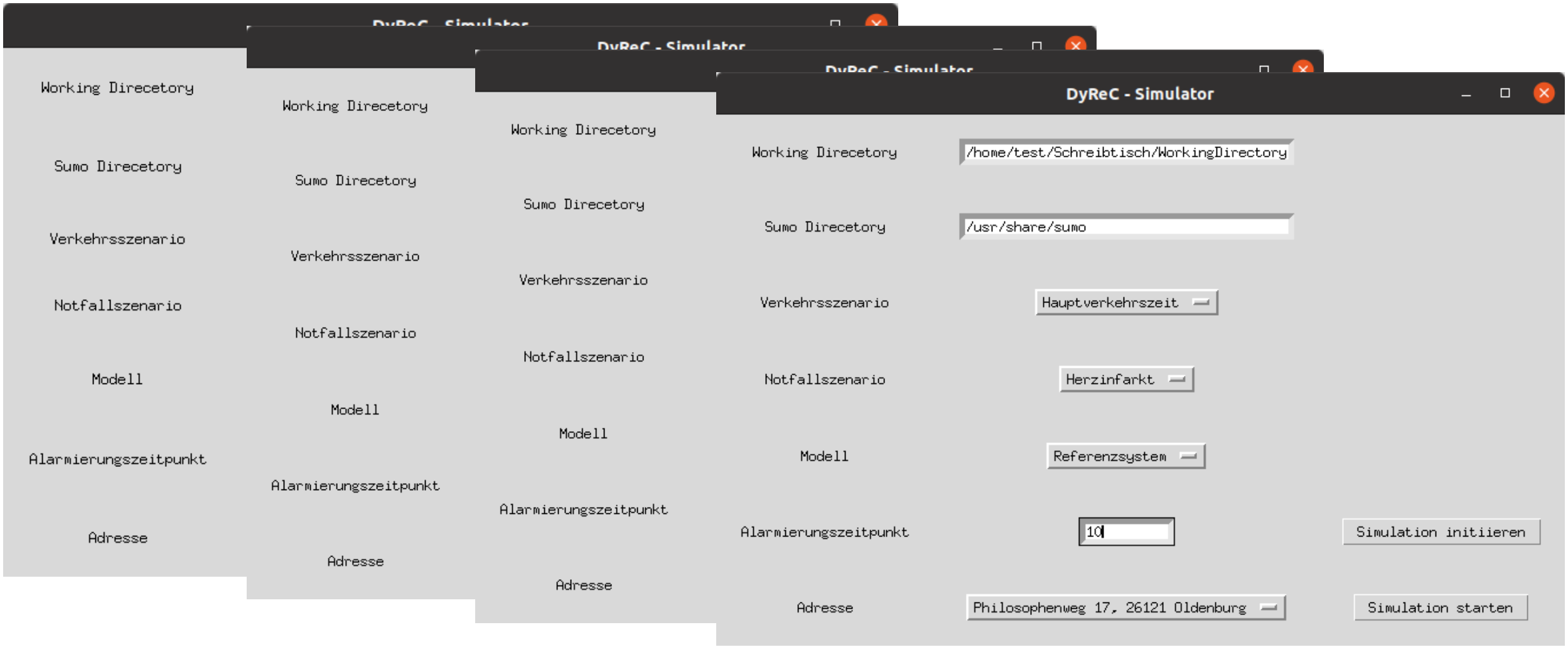
GUI


DyReC - Simulator
Working Direcetory
/home/test/Schreibtisch/WorkingDirectory
Sumo Direcetory
/usr/share/sumo
Verkehrsszenario
Verkehrsszenario waehlen
Notfallszenario
Notfallszenario waehlen
Modell
Modell waehlen
Alarmierungszeitpunkt
Simulation initiieren
Adresse
Adresse waehlen
Simulation starten

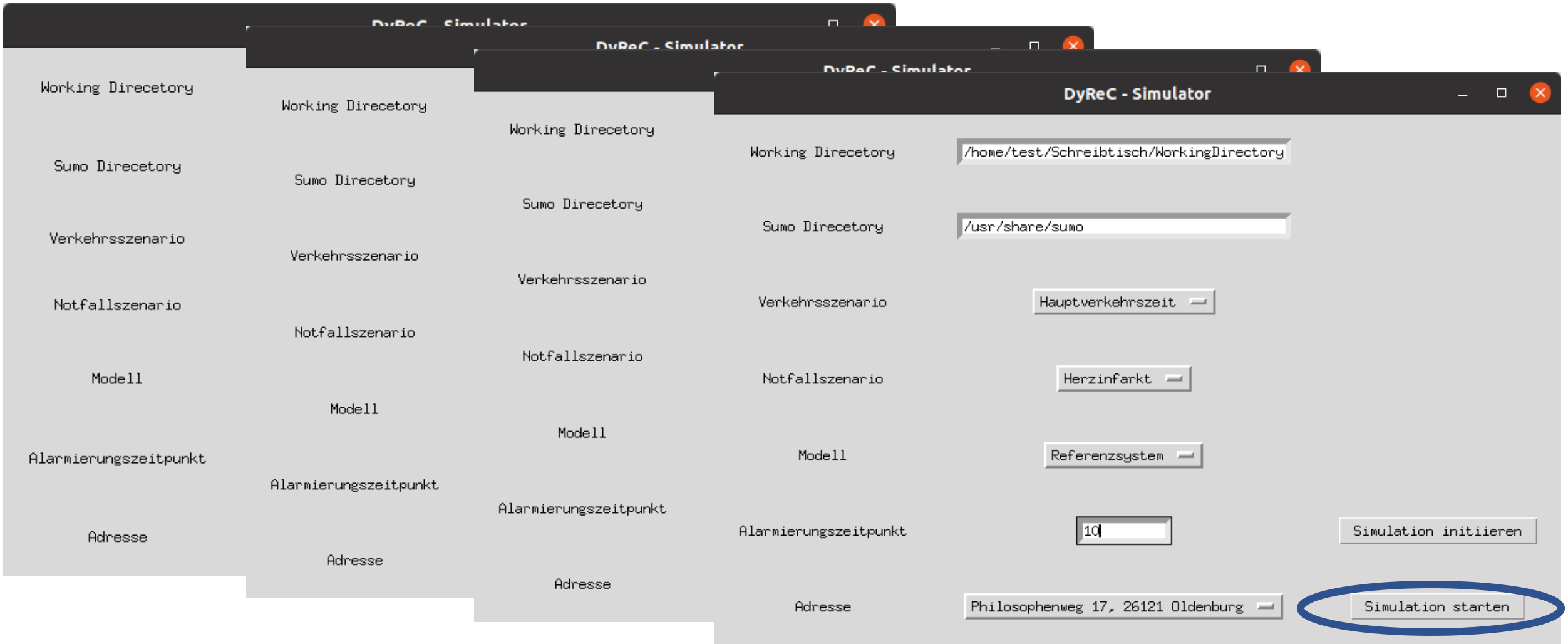
GUI


Working Directory
Sumo Directory
Verkehrsszenario
Notfallszenario
Modell
Alarmierungszeitpunkt
Adresse
Working Directory
Sumo Directory
Verkehrsszenario
Notfallszenario
Modell
Alarmierungszeitpunkt
Adresse
DyReC - Simulator
Working Directory
/home/test/Schreibtisch/WorkingDirectory
Sumo Directory
/usr/share/sumo
Verkehrsszenario
Verkehrsszenario waehlen
Hauptverkehrszeit
Nebenverkehrszeit
Schwachverkehrszeit
Notfallszenario
Notfallszenario waehlen
Modell
Modell waehlen
Alarmierungszeitpunkt
Simulation initiieren
Adresse
Adresse waehlen
Simulation starten

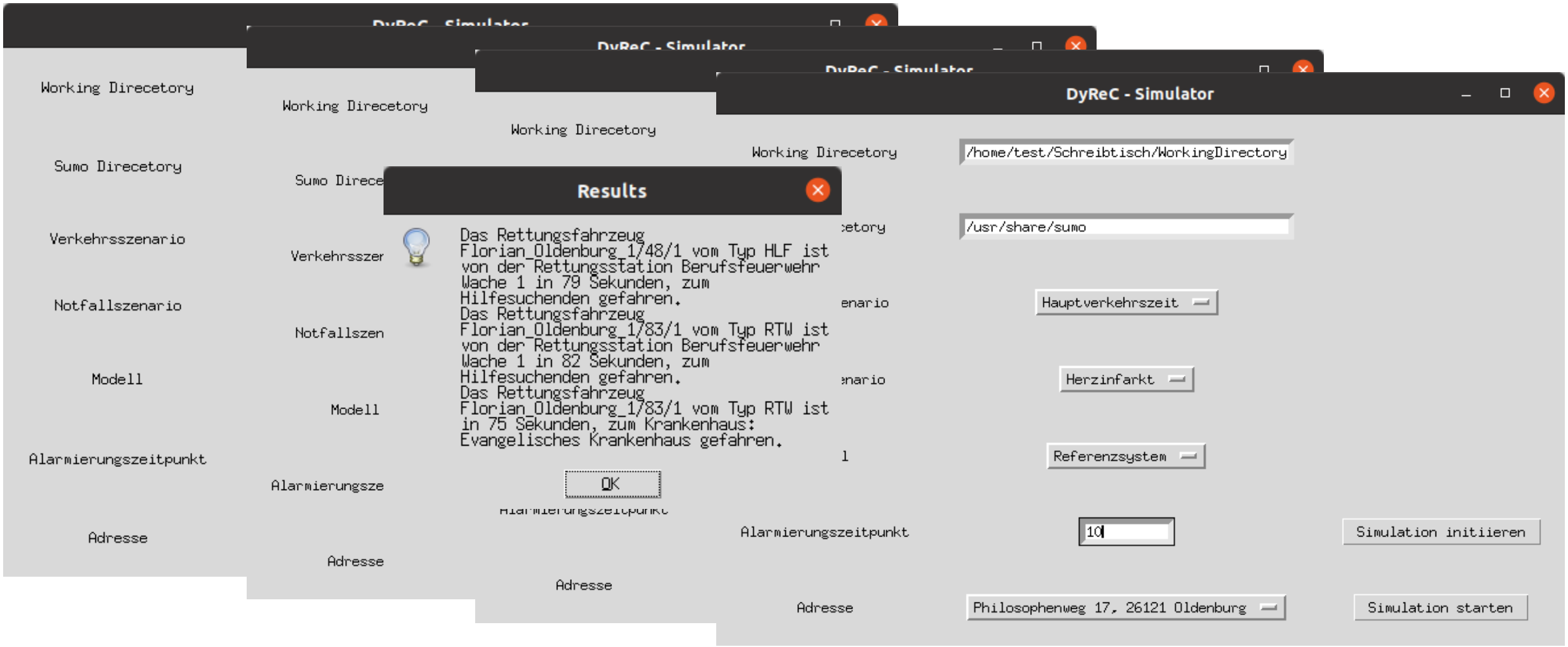
GUI


DyReC - Simulator
Working Direcetory
Sumo Direcetory
Verkehrsszenario
Notfallszenario
Modell
Alarmierungszeitpunkt
Adresse
DyReC - Simulator
Working Direcetory
Sumo Direcetory
Verkehrsszenario
Notfallszenario
Modell
Alarmierungszeitpunkt
Adresse
DyReC - Simulator
Working Direcetory
Sumo Direcetory
Verkehrsszenario
Notfallszenario
Modell
Alarmierungszeitpunkt
Adresse
DyReC - Simulator
Working Direcetory
/home/test/Schreibtisch/WorkingDirectory
Sumo Direcetory
/usr/share/sumo
Verkehrsszenario
Hauptverkehrszeit
Notfallszenario
Herzinfarkt
Modell
Referenzsystem
Alarmierungszeitpunkt
10
Simulation initiieren
Adresse
Philosophenweg 17, 26121 Oldenburg
Simulation starten

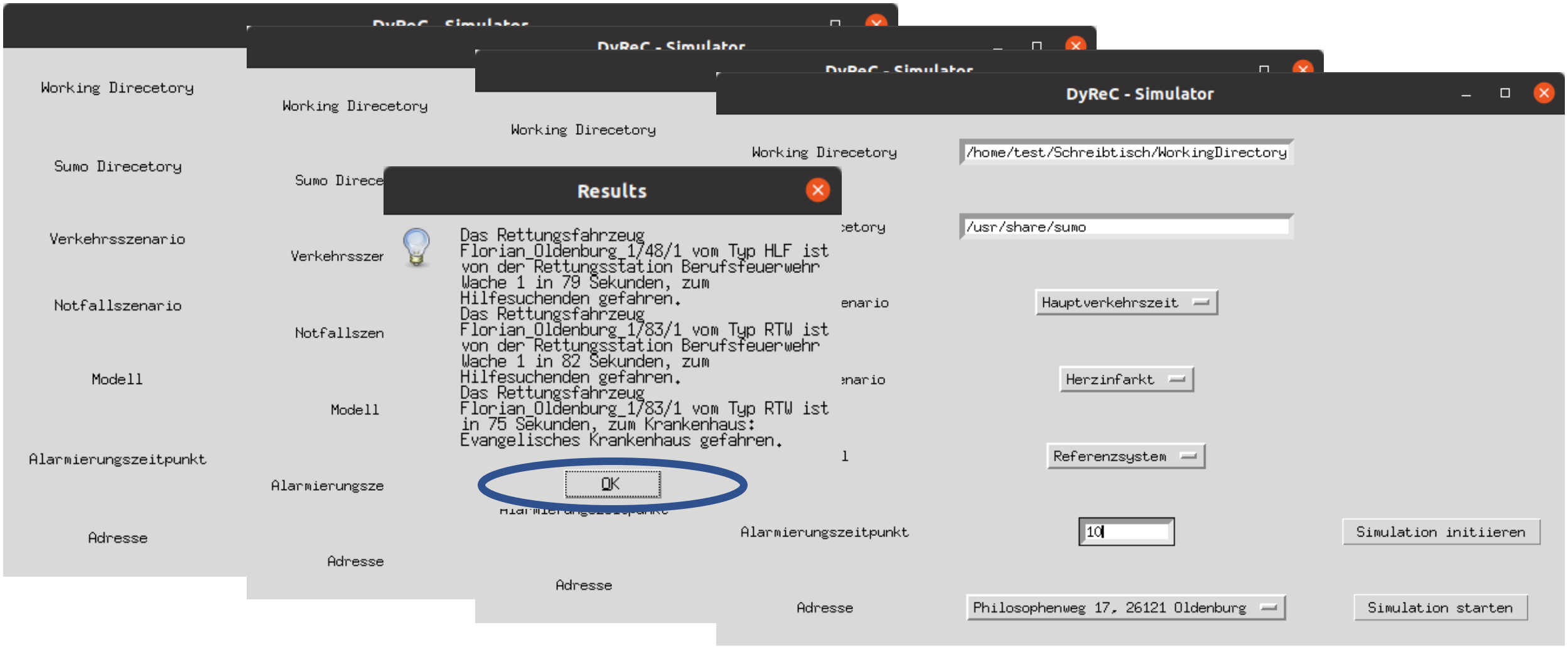
GUI


DyReC - Simulator
Working Directecory
Sumo Directecory
Verkehrsszenario
Notfallszenario
Modell
Alarmierungszeitpunkt
Adresse
DyReC - Simulator
Working Directecory
Sumo Directecory
Verkehrsszenario
Notfallszenario
Modell
Alarmierungszeitpunkt
Adresse
DyReC - Simulator
Working Directecory
Sumo Directecory
Verkehrsszenario
Notfallszenario
Modell
Alarmierungszeitpunkt
Adresse
DyReC - Simulator
Working Directecory
/home/test/Schreibtisch/WorkingDirectory
Sumo Directecory
/usr/share/sumo
Verkehrsszenario
Hauptverkehrszeit
Notfallszenario
Herzinfarkt
Modell
Referenzsystem
Alarmierungszeitpunkt
10
Simulation initiieren
Adresse
Philosophenweg 17, 26121 Oldenburg
Simulation starten

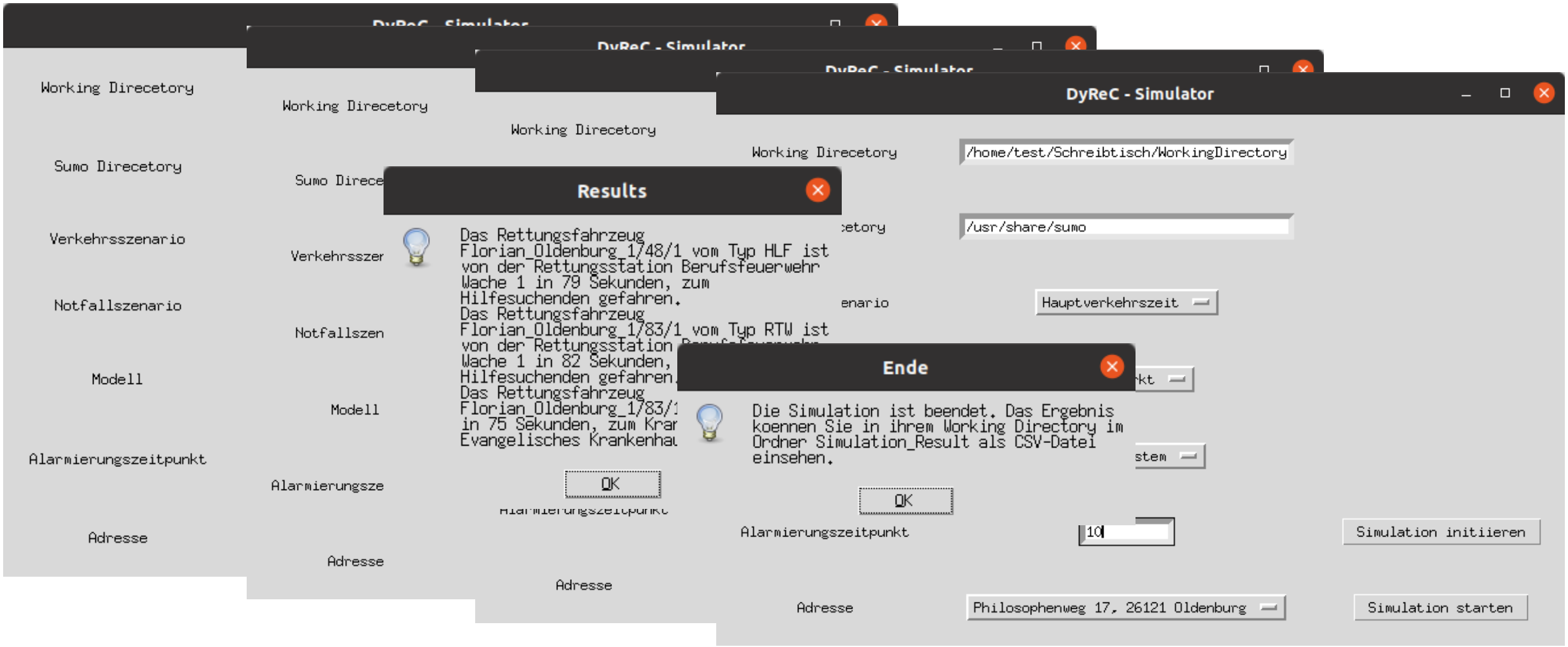
GUI


DyReC - Simulator
Working Directecory
Sumo Directecory
Verkehrsszenario
Notfallszenario
Modell
Alarmierungszeitpunkt
Adresse
Working Direcetory
Sumo Direce
Verkehrsszer
Notfallszen
Modell
Alarmierungsze
Adresse
Working Direcetory
Results
Das Rettungsfahrzeug Florian_Oldenburg_1/48/1 vom Typ HLF ist von der Rettungsstation Berufsfeuerwehr Wache 1 in 79 Sekunden, zum Hilfesuchenden gefahren.
Das Rettungsfahrzeug Florian_Oldenburg_1/83/1 vom Typ RTW ist von der Rettungsstation Berufsfeuerwehr Wache 1 in 82 Sekunden, zum Hilfesuchenden gefahren.
Das Rettungsfahrzeug Florian_Oldenburg_1/83/1 vom Typ RTW ist in 75 Sekunden, zum Krankenhaus: Evangelisches Krankenhaus gefahren.
OK
Adresse
Working Direcetory
/home/test/Schreibtisch/WorkingDirectory
/usr/share/sumo
Hauptverkehrszeit
Herzinfarkt
Referenzsystem
Alarmierungszeitpunkt
10
Simulation initiieren
Adresse
Philosophenweg 17, 26121 Oldenburg
Simulation starten

GUI


DyReC - Simulator
Working Directecory
Sumo Directecory
Verkehrsszenario
Notfallszenario
Modell
Alarmierungszeitpunkt
Adresse
Working Direcetory
Sumo Direce
Verkehrsszer
Notfallszen
Modell
Alarmierungsze
Adresse
Working Direcetory
Results
Das Rettungsfahrzeug Florian_Oldenburg_1/48/1 vom Typ HLF ist von der Rettungsstation Berufsfeuerwehr Wache 1 in 79 Sekunden, zum Hilfesuchenden gefahren.
Das Rettungsfahrzeug Florian_Oldenburg_1/83/1 vom Typ RTW ist von der Rettungsstation Berufsfeuerwehr Wache 1 in 82 Sekunden, zum Hilfesuchenden gefahren.
Das Rettungsfahrzeug Florian_Oldenburg_1/83/1 vom Typ RTW ist in 75 Sekunden, zum Krankenhaus: Evangelisches Krankenhaus gefahren.
OK
Adresse
DyReC - Simulator
Working Direcetory
/home/test/Schreibtisch/WorkingDirectory
/usr/share/sumo
Hauptverkehrszeit
Herzinfarkt
Referenzsystem
Alarmierungszeitpunkt
10
Simulation initiieren
Adresse
Philosophenweg 17, 26121 Oldenburg
Simulation starten

GUI


DyReC - Simulator
Working Directecory
Sumo Directecory
Verkehrsszenario
Notfallszenario
Modell
Alarmierungszeitpunkt
Adresse
Working Directecory
Sumo Direce
Verkehrsszer
Notfallszen
Modell
Alarmierungsze
Adresse
Working Directecory
Results
Das Rettungsfahrzeug Florian_Oldenburg_1/48/1 vom Typ HLF ist von der Rettungsstation Berufsfeuerwehr Wache 1 in 79 Sekunden, zum Hilfesuchenden gefahren.
Das Rettungsfahrzeug Florian_Oldenburg_1/83/1 vom Typ RTW ist von der Rettungsstation
Wache 1 in 82 Sekunden, Hilfesuchenden gefahren.
Das Rettungsfahrzeug Florian_Oldenburg_1/83/1
in 75 Sekunden, zum Krar
Evangelisches Krankenhau
OK
Adresse
DyReC - Simulator
Working Directecory
/home/test/Schreibtisch/WorkingDirectory
/usr/share/sumo
Hauptverkehrszeit
Ende
Die Simulation ist beendet. Das Ergebnis koennen Sie in ihrem Working Directory im Ordner Simulation_Result als CSV-Datei einsehen.
OK
Alarmierungszeitpunkt
10
Simulation initiieren
Adresse
Philosophenweg 17, 26121 Oldenburg
Simulation starten