

Fortgeschrittenenprojekt Wirtschaftsinformatik (SS 2016)

Simulation des Fahrverhaltens an einer
Ampelkreuzung

Nina Trilck, Nikita Maslov, Tran Khac Dat

22.09.2016

Gliederung

- 1. Einleitung
- 2. Lösungsideen
- 3. Umsetzung
- 4. Evaluation
- 5. Fazit

1. Einleitung

A

- Das Thema, das wir gewählt haben

B

- Die Motivation für dieses Thema

C

- Die Fragen, die mit diesem Thema beantwortet werden können

A. Das Thema, das wir gewählt haben

- Simulation des Fahrverhaltens an einer Ampelkreuzung

B. Die Motivation für dieses Thema

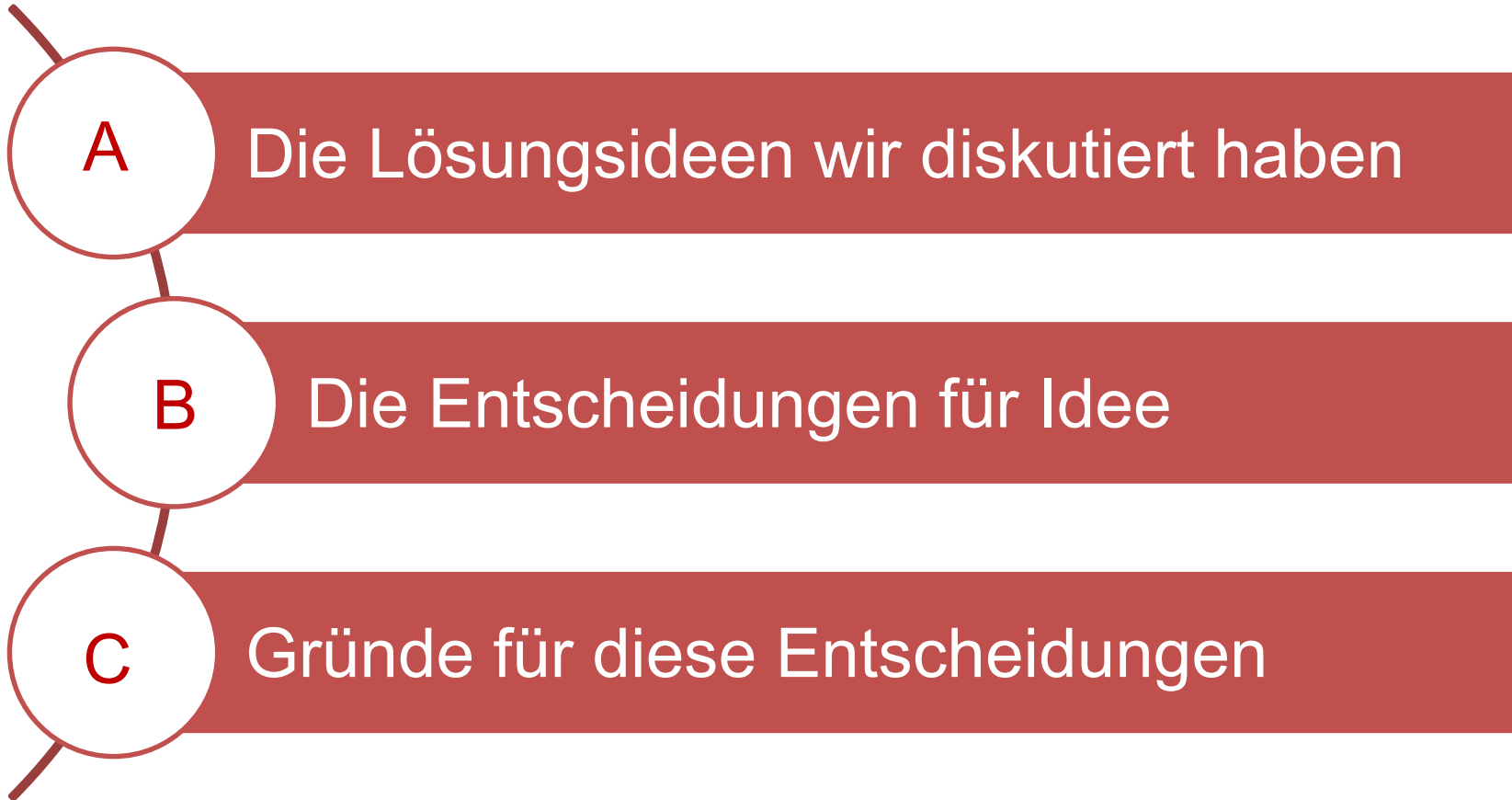
- Vorerst Grundmodell
- Stau ist ein Problem
- Anhand der Simulation
Bewertung potenzieller
Stausituationen
- Erweitern die Funktionen in der Zukunft wie
z.B: Anzahl der Autos, Parameter von
Geschwindigkeit eingeben



C. Die Fragen, die mit diesem Thema beantwortet werden können

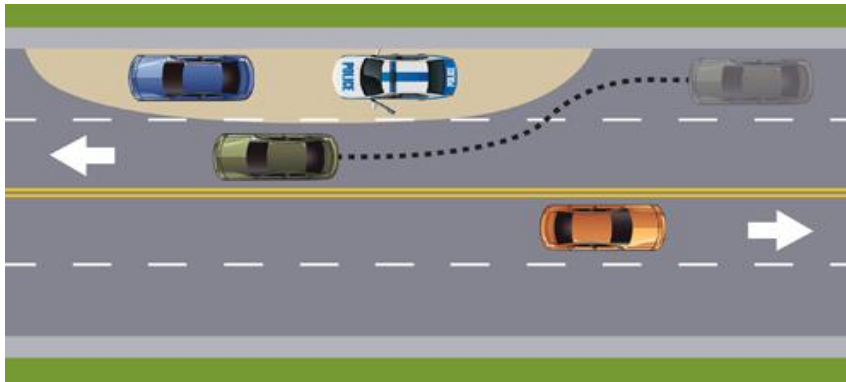
- Wie ist Verhalten der Fahrzeuge in einer Kreuzung?
- Wann kann ein Stau auftauchen?
- Wovon ist ein Stau abhängig?

2. Lösungsideen



A. Die Lösungsideen wir diskutiert haben

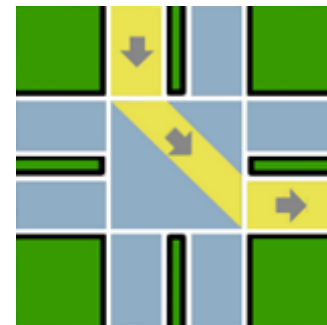
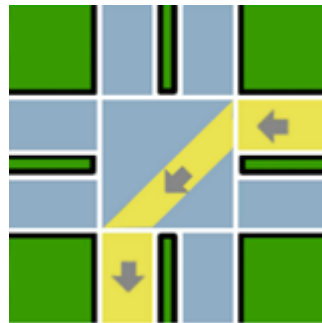
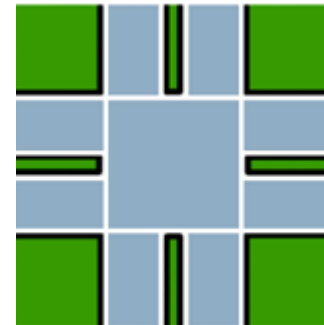
- Car-following model (Nagel-Schreckenberg Modell)
- Car lane changing model



www.servicenl.gov.nl.ca

- Multi-Threading, ThreadPool
- Java Swing, um Grafik zu darstellen
- Libgdx, um Grafik zu darstellen

Erste Ideen über Fahrverhalten



B. Die Entscheidungen für Idee

- Multi-Threading, ThreadPool
- Libgdx, um Grafik zu darstellen
- Car-following model (Nagel-Schreckenberg Modell)

C. Gründe für diese Entscheidungen

- Multi-Threading, ThreadPool
- Java Swing ist kompliziert zum Verwenden und hat eine schlechte Wirkung auf die Laufzeit -> Wählen Libgdx aus.
- Car-following model (Nagel-Schreckenberg Modell)

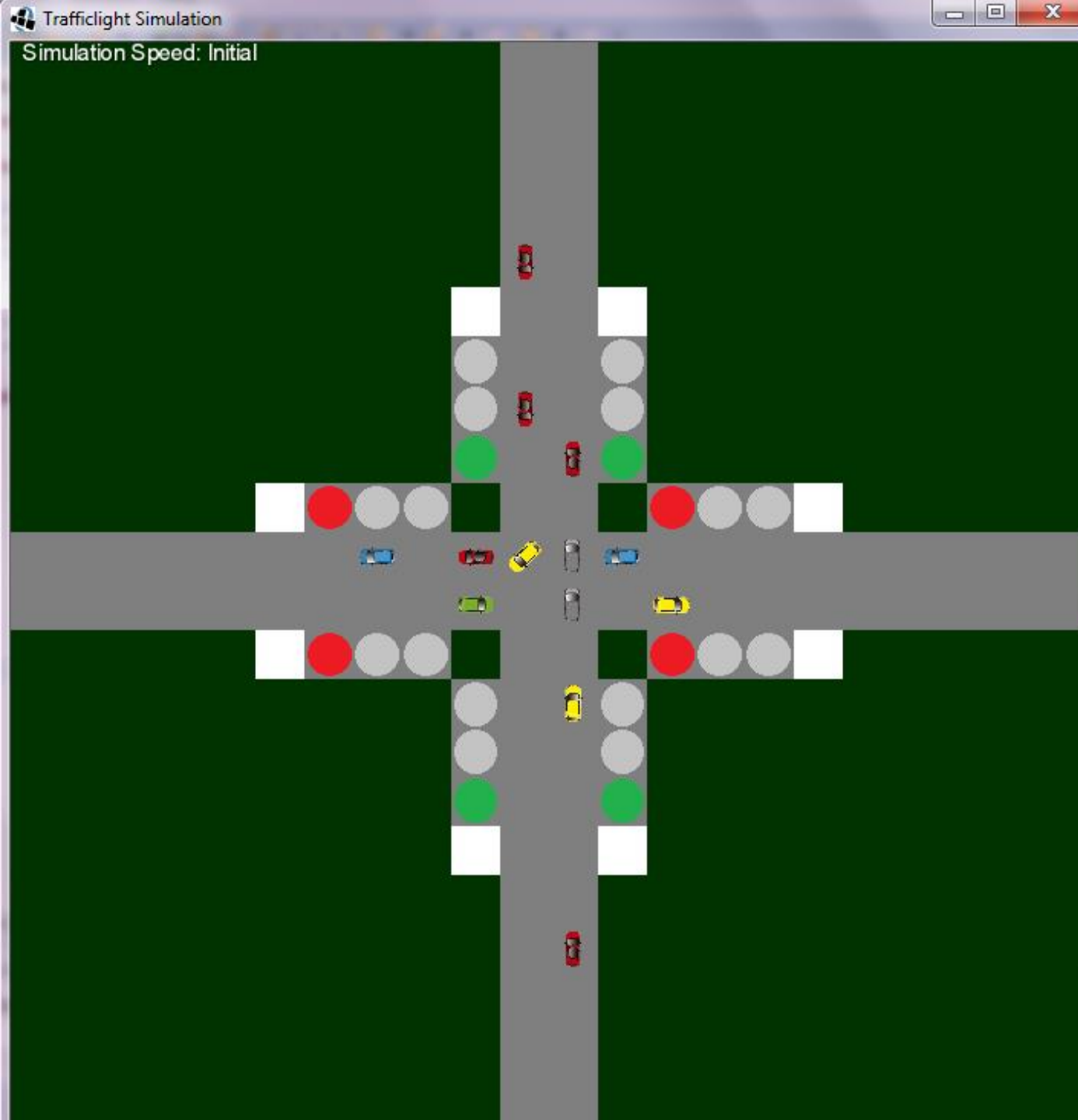
3. Umsetzung

A. Die Umsetzung
(Algorithmus) der Idee

B. Die Probleme während der
Umsetzung

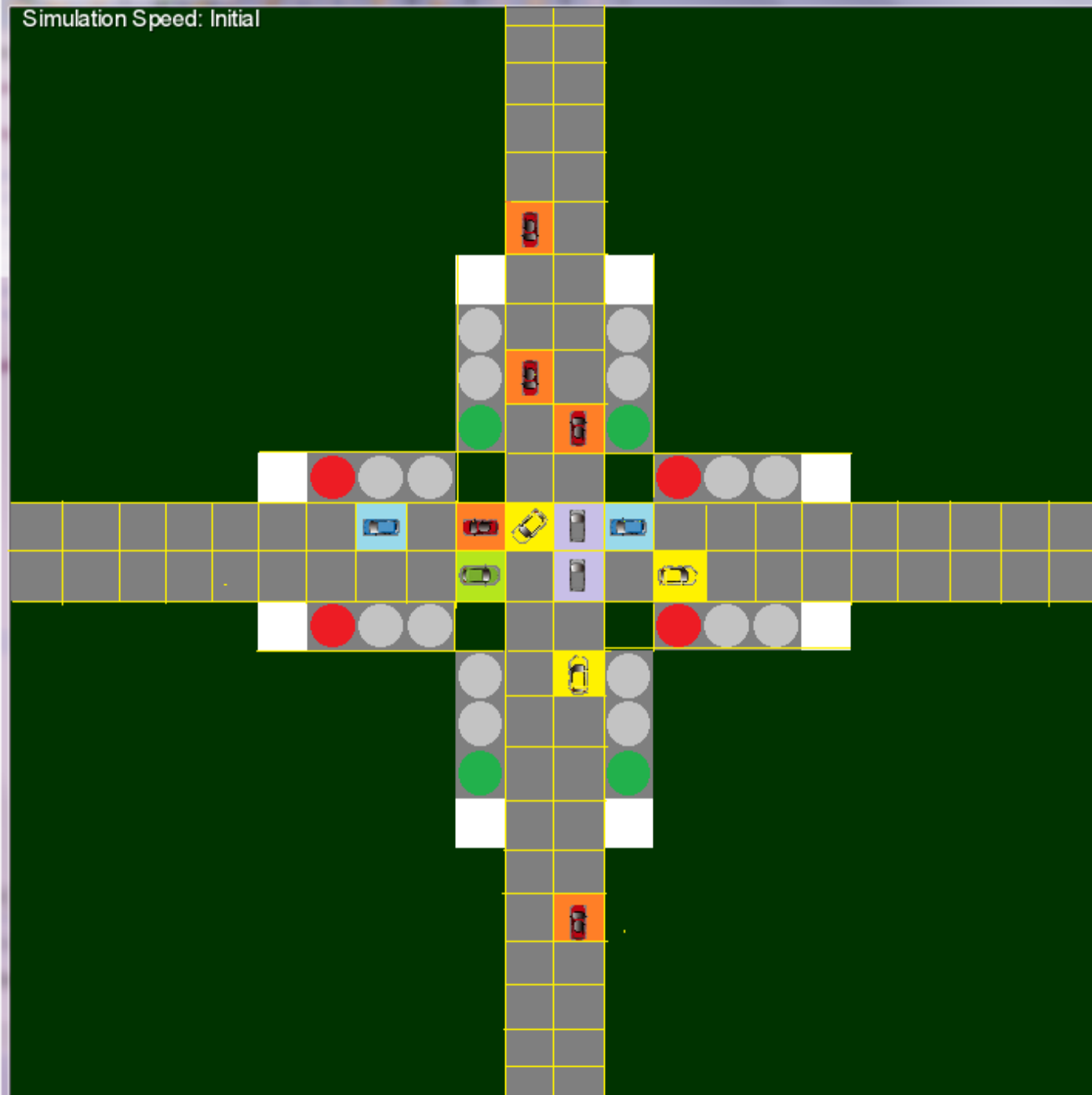
A. Die Umsetzung (Algorithmus) der Idee

- Erster Überblick
- Weitere Erklärungen und Demo ausführen



Erstes Screenshot

Simulation Speed: Initial



RedCarSouth



BlueCarWest



GreenCarEast



YellowCarSouth
West

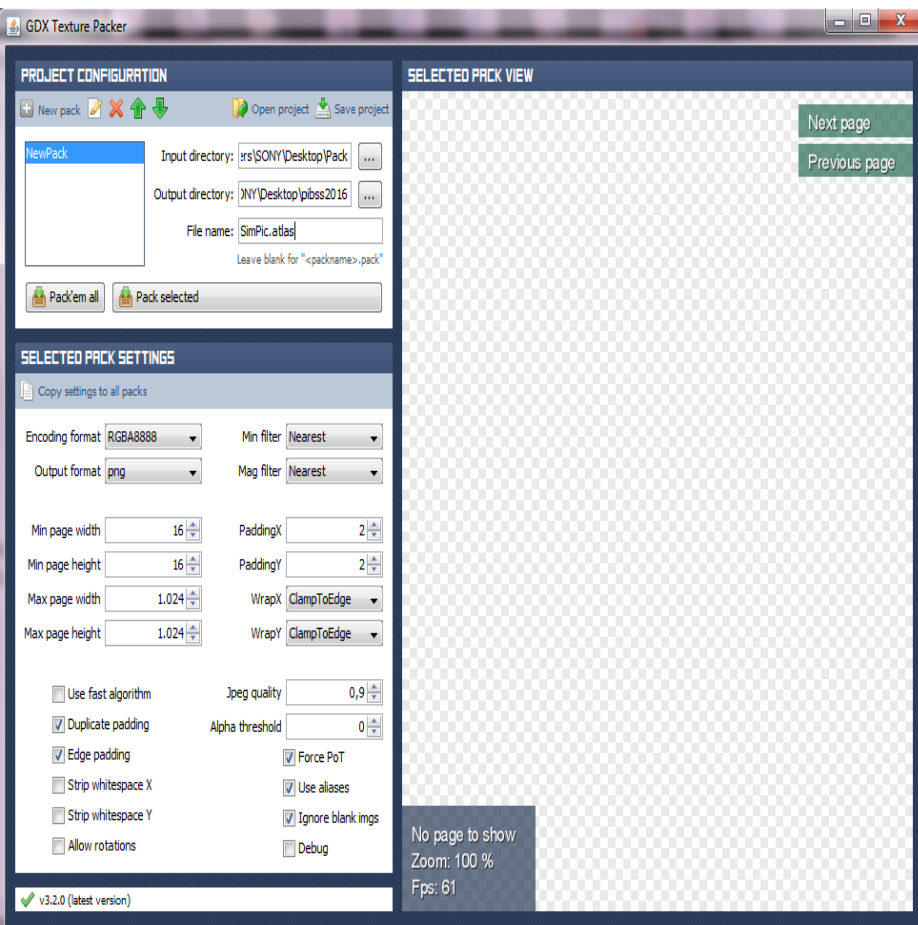


Street

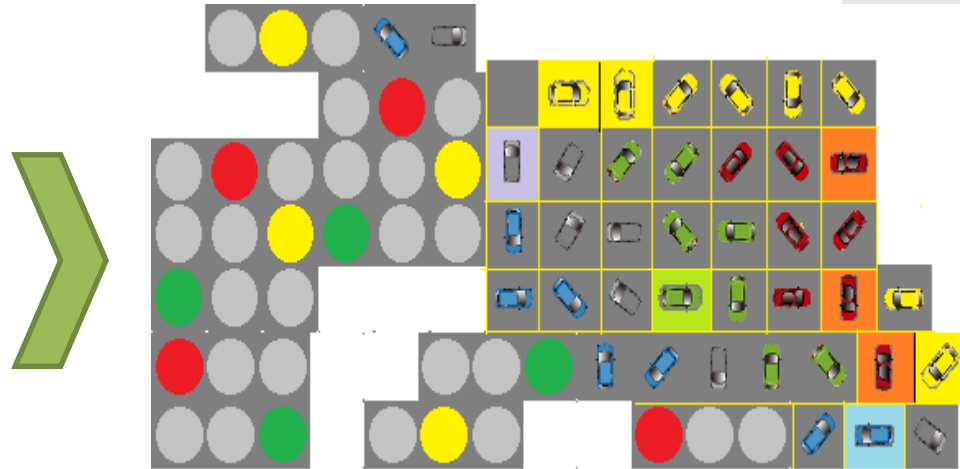


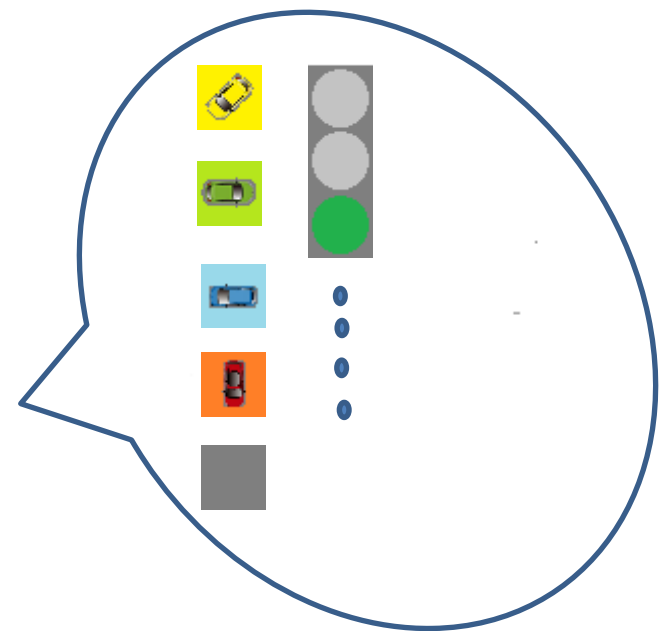
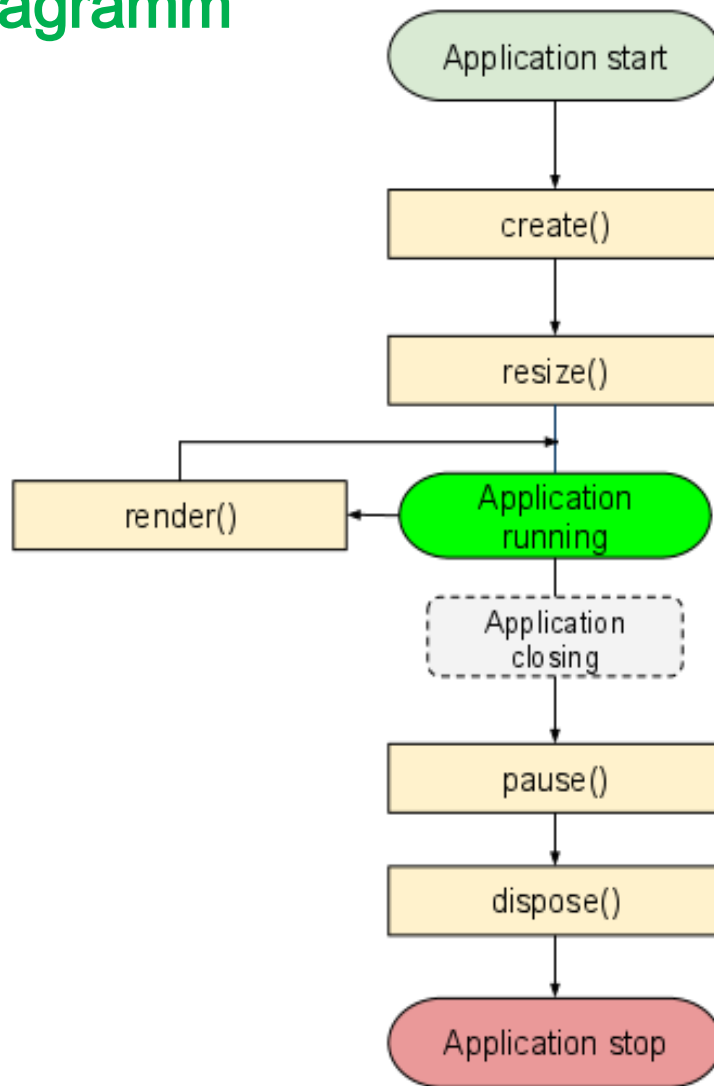
NorthTrafficGreen

Texture Packer



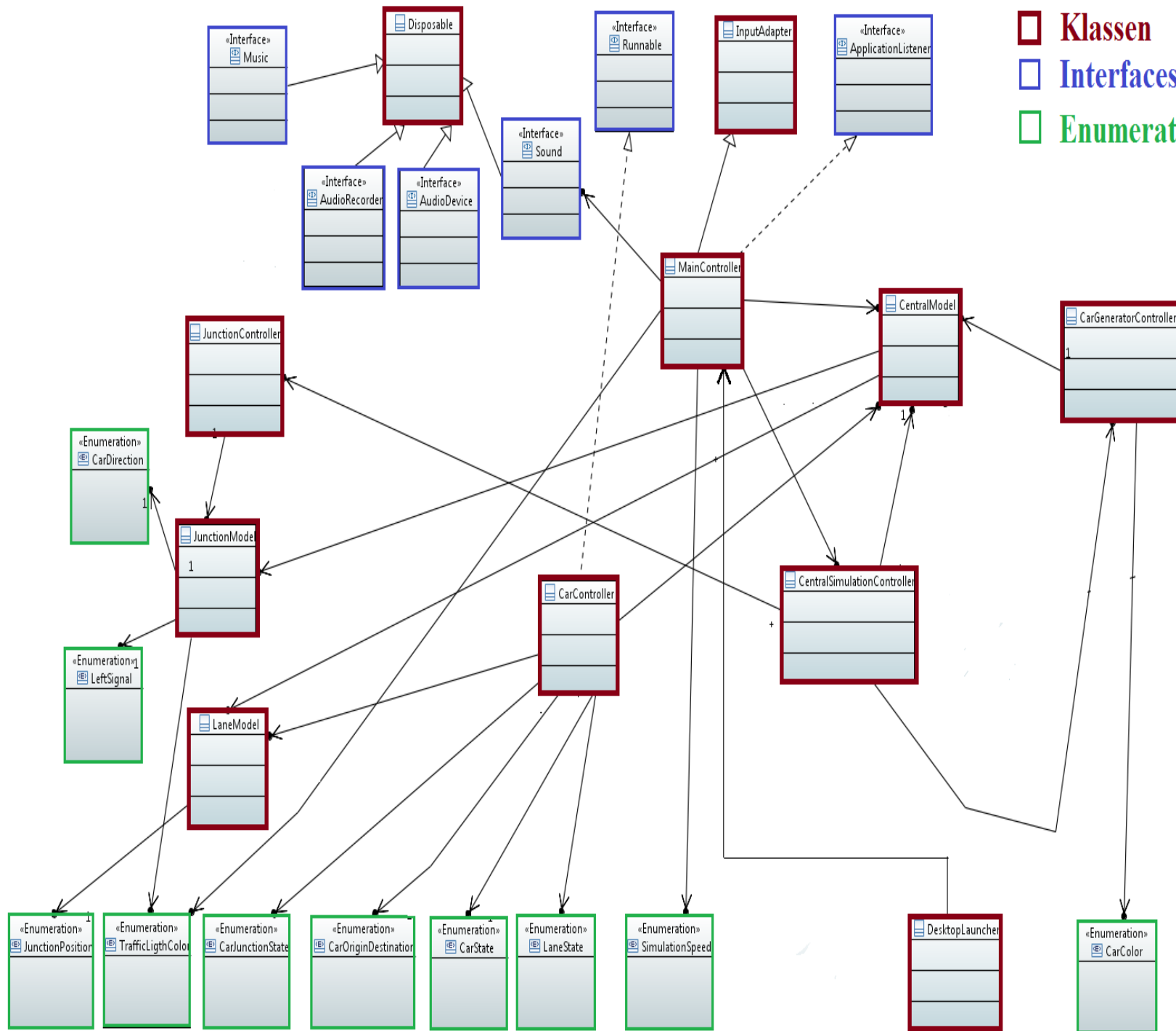
SimulationPicture.atlas





Weitere Erklärungen und Demo ausführen

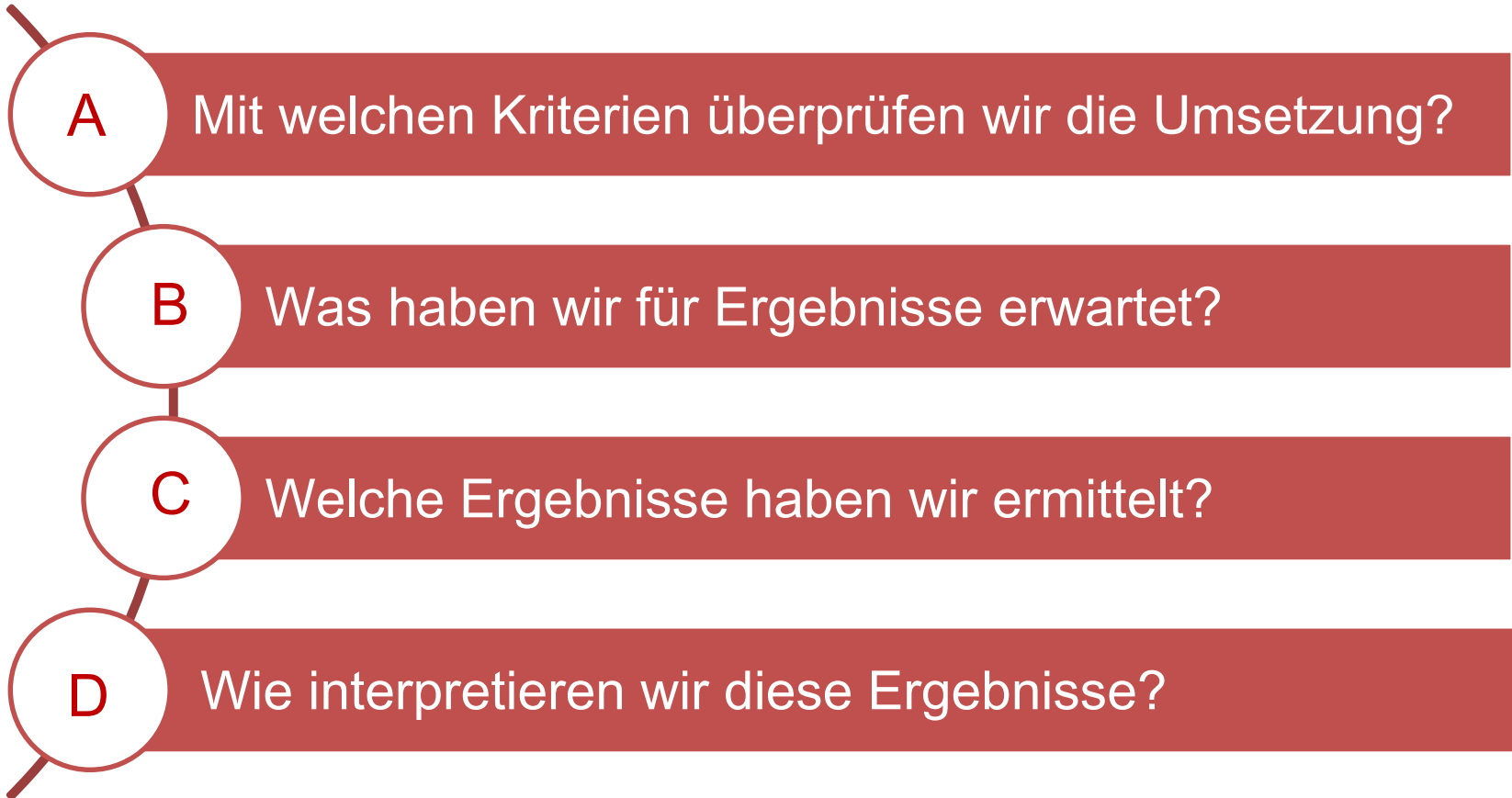
- Vereinfachendes Klassendiagramm
- Demo ausführen



B. Die Probleme während der Umsetzung

- Multi-Threading Problem
- Abbiegen-Verhalten Problem
- Kollisionen-Vermeidung zwischen Autos Problem
- Factory-Method zum Erstellen Problem
- Leistung von Programm

4. Evaluation



- **A. Mit welchen Kriterien überprüfen wir die Umsetzung?**
 - Änderung der Konfigurationseigenschaften:
 - SpawnRate
 - LaneLength
 - WaitingTime
- **B. Was haben wir für Ergebnisse erwartet?**
 - Volle/ korrekte Funktionsweise der Simulation bleibt erhalten
- **C. Welche Ergebnisse haben wir ermittelt?**
 - Änderung der eingegebenen Werte beeinträchtigen die Simulation nicht
- **D. Wie interpretieren wir diese Ergebnisse?**
 - Algorithmus der Simulation wurde effektiv umgesetzt

5. Fazit

A

- Dinge, die wir persönlich gelernt haben

B

- Was würden wir beim nächsten Mal anders machen?

A. Dinge, die wir persönlich gelernt haben

- Realitätsnahes Arbeiten
- Neue Erkenntnisse gewonnen: Github, Maven, LibGDX, Threads, YAML...
- Projektorganisation

B. Was würden wir beim nächsten Mal anders machen?

- Es könnten die neuen Funktionen wie Kontrolle der Anzahl der Autos, Tempo der Autos messen, Stau vorwarnen... ergänzt werden
- Es könnten intelligente Ampeln simuliert werden

Vielen Dank für Ihre Aufmerksamkeit