

# Mötesanteckningar



GÖTEBORGS  
UNIVERSITET

**CHALMERS**

Gustaf Ringius

April 3, 2014

Dagordning:<sup>1</sup>

## 1. Restrospective

- Vad har gått bra?
  - Coren blev klar och presenterad
  - GUI ifrån endast en canvas blir bra.
  - Mötestiderna har hållts. Alla har kommit och varit i tid.
  - Arbetsfördelningen har fungerat bättre.
  - Inga cirkulära beroenden.
  - 100 procent test coverage på trafficentity, vehicle, car & truck.
  - Ny analysmodell har tagits fram och RAD i princip helt klar.
  - Implementeringstiderna fungerar väl.
- Vad har gått dåligt?
  - Robert förvirrad över klassindelning med Andreas
  - Kod och dokumentationsslarv. Saknade annotationer och javadoc.
  - Avsaknad av tester, kanske ha en person som får tester som ansvar?
  - Har inte börjat på SDD som utlovades i måndags.
  - Försöka få till mindre metoder / bättre uppdelning av logik så det blir trivialt att skriva effektivare test och göra koden mer lättläst och lättbegriplig samt undvika redundant kod.
  - Fler mindre test ger enklare debugging.

## 2. Lägga till musikbibliotek.

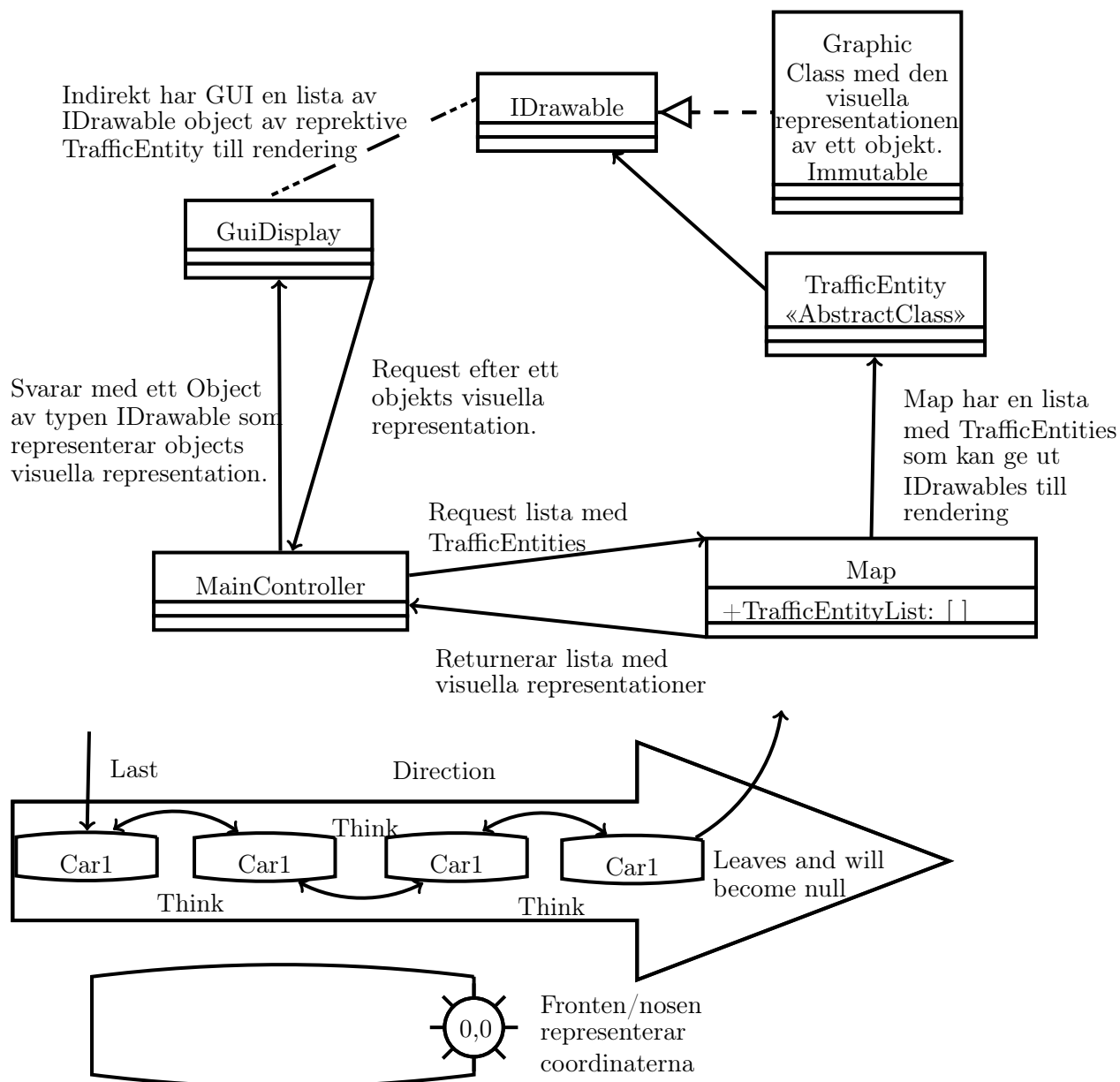
## 3. Components vs inheritance. Bilar ska endast ha sitt eget tänkade resten bör samlas i vehicle eller annan superklass.

---

<sup>1</sup>Närvarande: Gustaf Ringius, Andreas Löfman, Robert Wennergren, Felix Barring

#### 4. Bilars grafiska representation.

- Två trådar, en för logik och en för grafisk representation.
- Interface paintable, GUI ska inte veta vad den ritar.
- En bil har ett objekt av t ex en graphics-klass som innehåller en visuell representation av TrafficEntityn. När guit vill rita en Entity så ropar den på TrafficEntity.getGraphics() som är synkroniserad och returnerar ett graphics-objekt som sedan endast guit indirekt kan ändra i vilket gör den visuella delen trådsäker. Graphic-objektet är en kopia av bilens ritbara state.
- Bilens position ligger i nosen på bilens visuella representation.



5. Noderna bör vara final för då blir dessa trådsäkra vid ritning i guit.
6. Testa guit med bot bör ej vara relevant då det indikerar felaktig MVC.
7. Implementering till nästa möte
  - Gustaf & Felix: IPaintable och bilens grafikrepresentation. DIA av våra implementeringsval samt ytterligare en abstraktionsnivå med en entity som TrafficEntity ärver ifrån för att kunna ha byggnader och andra entiteter som inte specifikt är trafikentiteter.
  - Andreas: Upptagen tills nästa möte (Golfresa) men ska fundera på lösning till noders final koordinater för trådsäkerhet till guit.
  - Robert: Fortsätter med sina noder.
8. Nästa sprint
  - Grafisk representation av bilar som kör mellan A och B. Gärna med korsning.
  - Göra SDD
  - Noder ska vara funktionella. Randomisering av vilken bil som kör först om det är kö i alla riktningar. Högerregeln.
  - Påbörjande av algorithm för högerregeln beroende på vinkeln.
9. Nästa möte
  - Inspektera Gustaf & Felix gui-koppling
  - Inspektera Roberts noder.
  - Höra Andreas lösning på trådsäkerheten hos noder.
  - Inspektera Andreas SDD-grund.
  - Dela ut ytterligare arbete tydligare för enklare parallel utveckling.