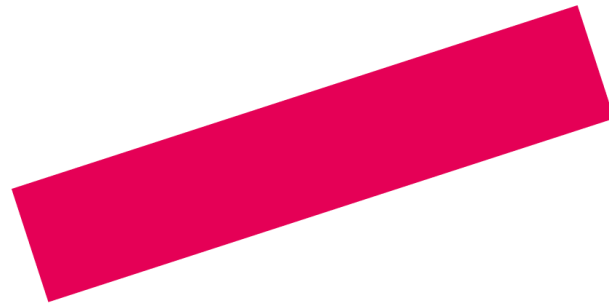


HBO-ICT Embedded Software Development

# Real-Time Use Cases

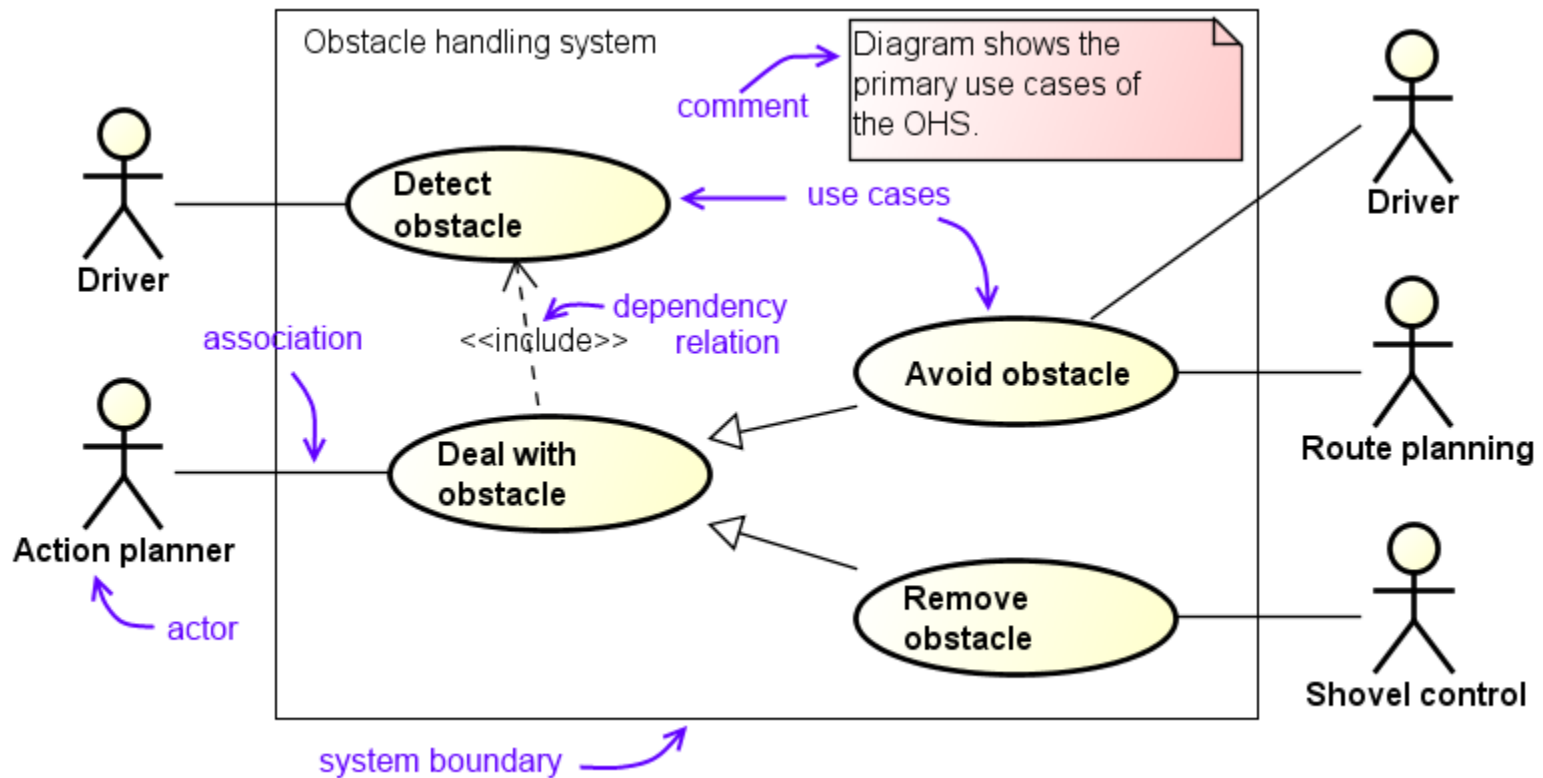


World of Robots :: World

# Lesdoelen

- Het kunnen benoemen van een usecase voor een embedded systeem.
- Het kunnen aanwijzen van subsystemen die een rol spelen in de usecase.
- Het kunnen vastleggen van de usecase met behulp van UML.
- Het kunnen opdelen van complexe usecases in deelusecases.

# Use Case diagram (korte herhaling)



# Use Case diagram

Use case diagrams worden gebruikt tijdens de **analyse** van een systeem:

- Een **use case** definieert een functionele mogelijkheid van het systeem.
- Een **actor** is een object buiten het systeem dat een interactie met het systeem aangaat.
  - Berichten worden heen en/of terug gestuurd tijdens het uitvoeren van de use case.

Let op:

- Het gaat hier om zichtbare mogelijkheden en resultaten van het systeem.  
**Niet zichtbaar van buiten het systeem → geen use case.**
  - Dus geen onderdeel van de requirements analyse, maar iets voor het ontwerp.
- Een use case zegt niets over een ontwerp of implementatie van het systeem.

# Benoemen van Use Cases

Benoem de actoren en use cases voor het volgende systeem:

- Airbag in een auto

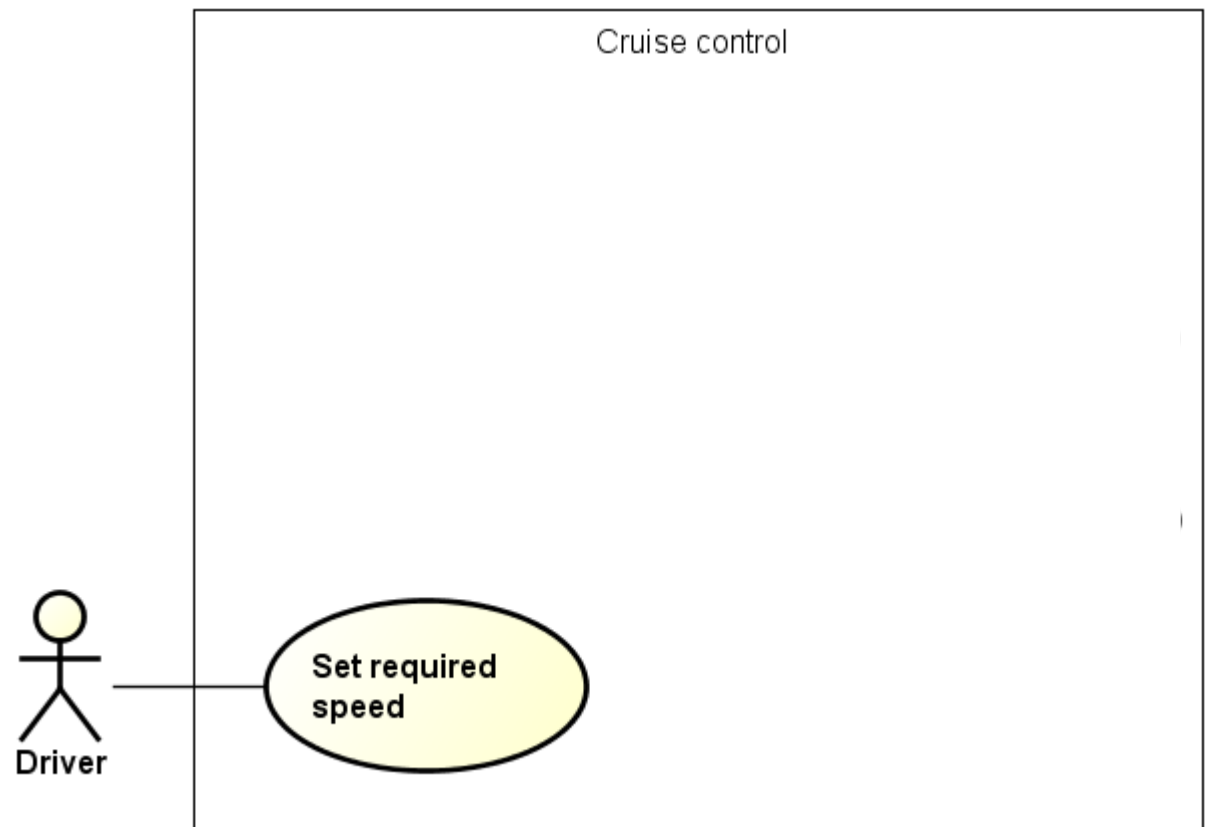
# Benoemen van Use Cases

Bepaal de actoren en use cases voor het volgende systeem:

- Thermostaatgeregelde verwarming

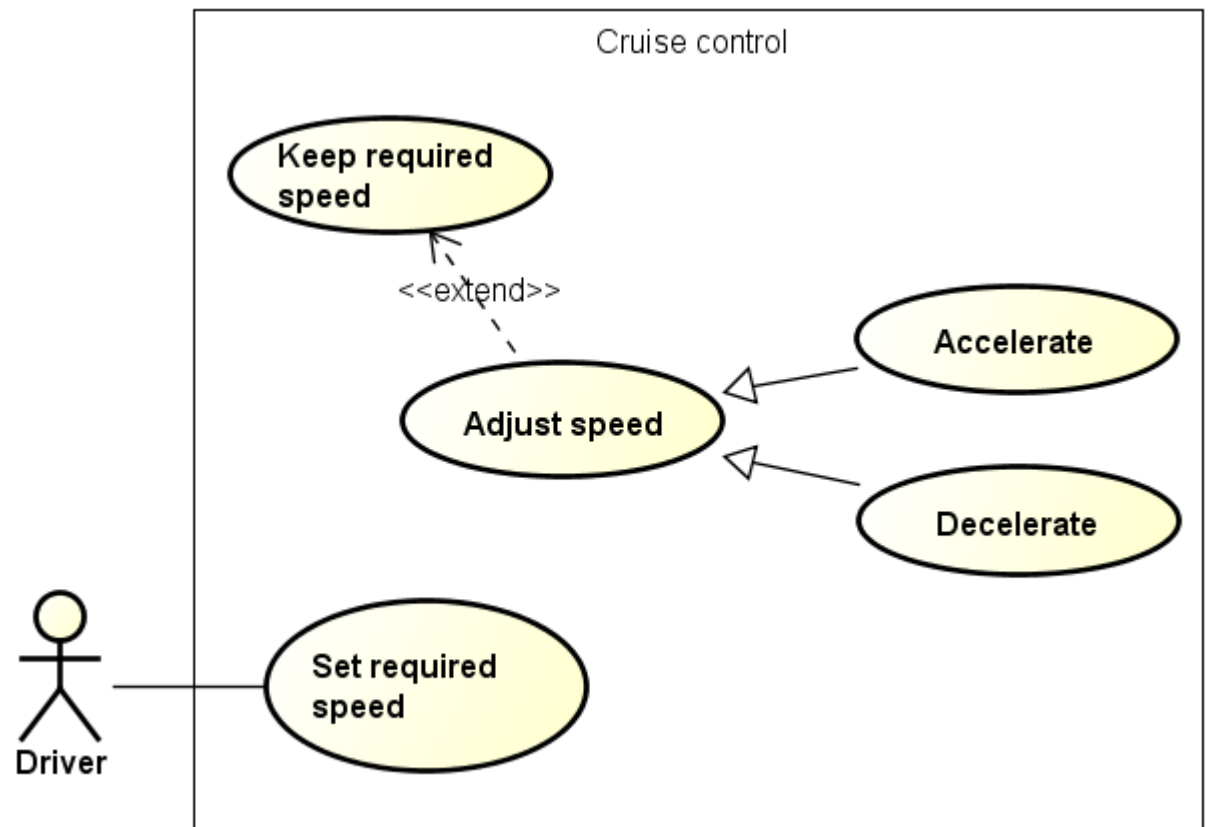
# Voorbeeld Subsystemen (cruise control)

Voorbeeldsysteem  
voor het opdelen in  
subsystemen.



# Voorbeeld Subsystemen (cruise control)

Voorbeeldsysteem  
voor het opdelen in  
subsystemen.

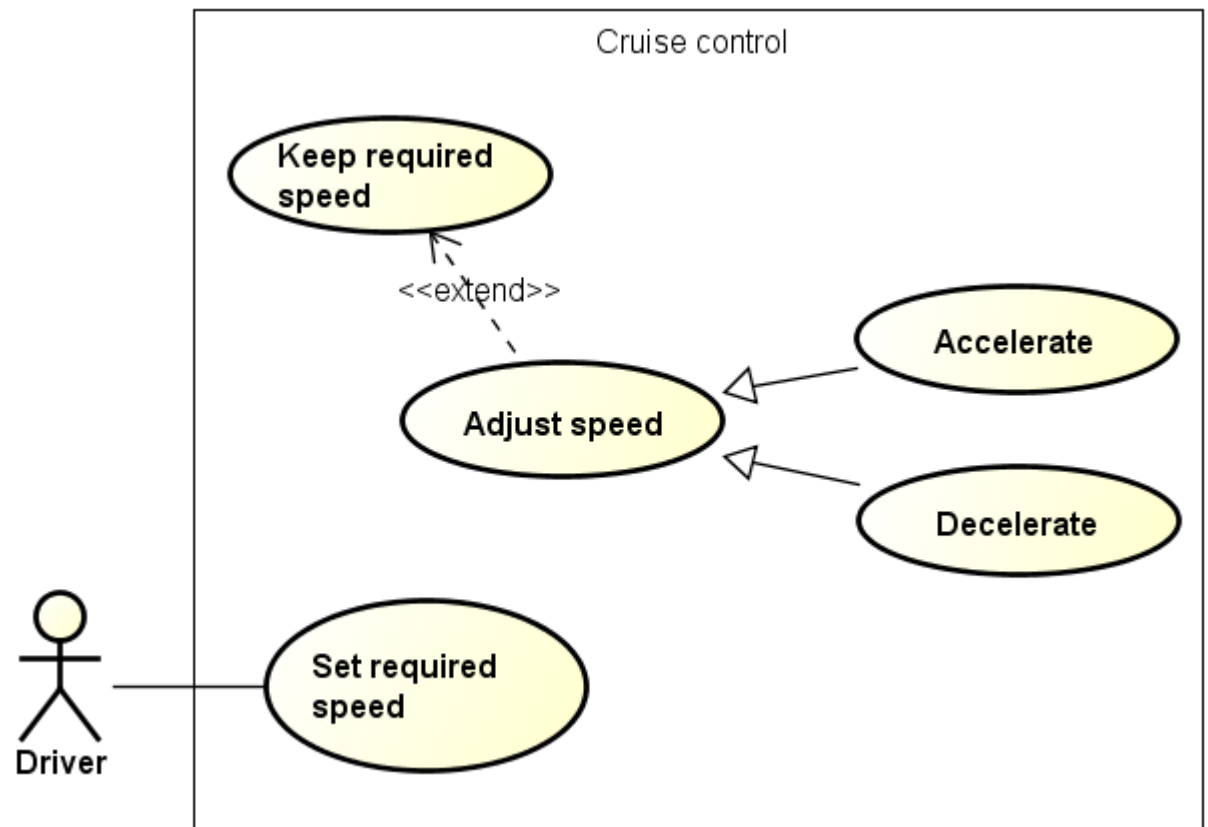




# Voorbeeld Subsystemen (cruise control)

Voorbeeldsysteem  
voor het opdelen in  
subsystemen.

Welke subsystemen  
spelen hier  
waarschijnlijk een  
rol?



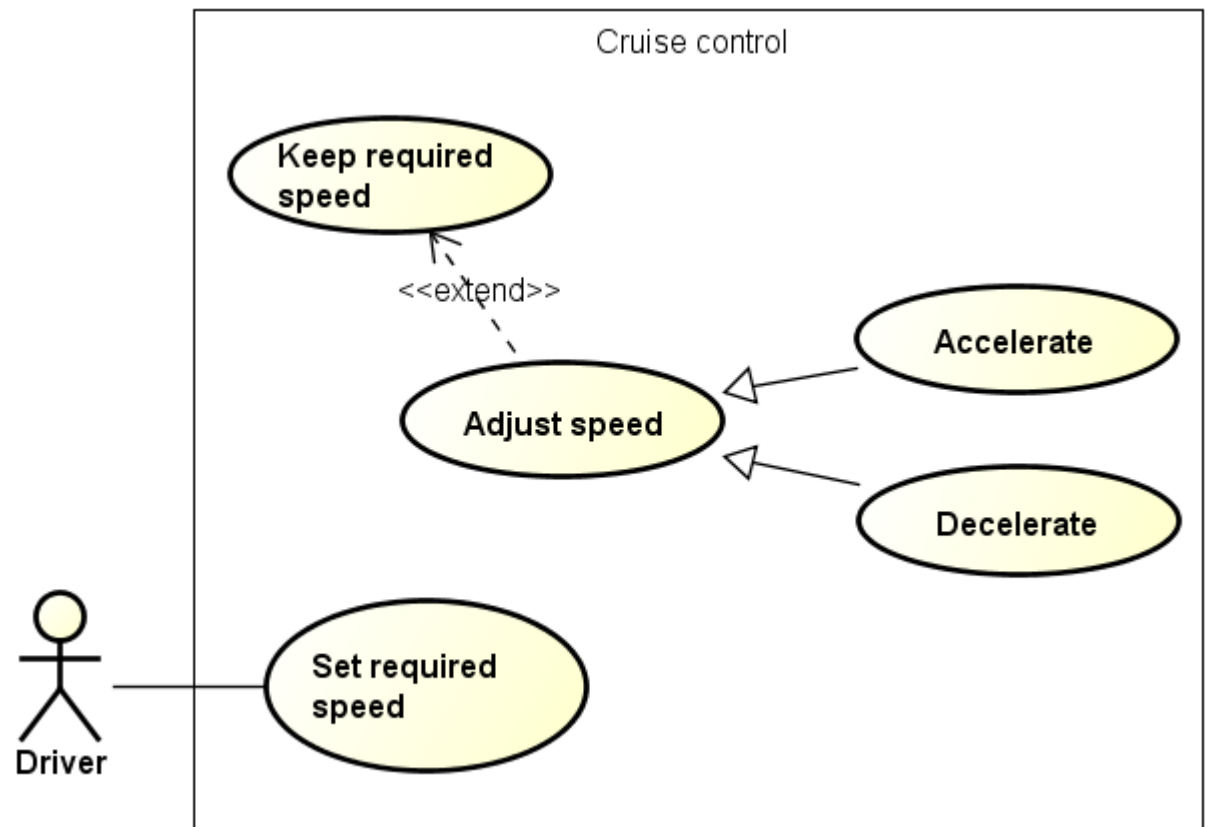
# Voorbeeld Subsystemen (cruise control)

Voorbeeldsysteem voor het opdelen in subsystemen.

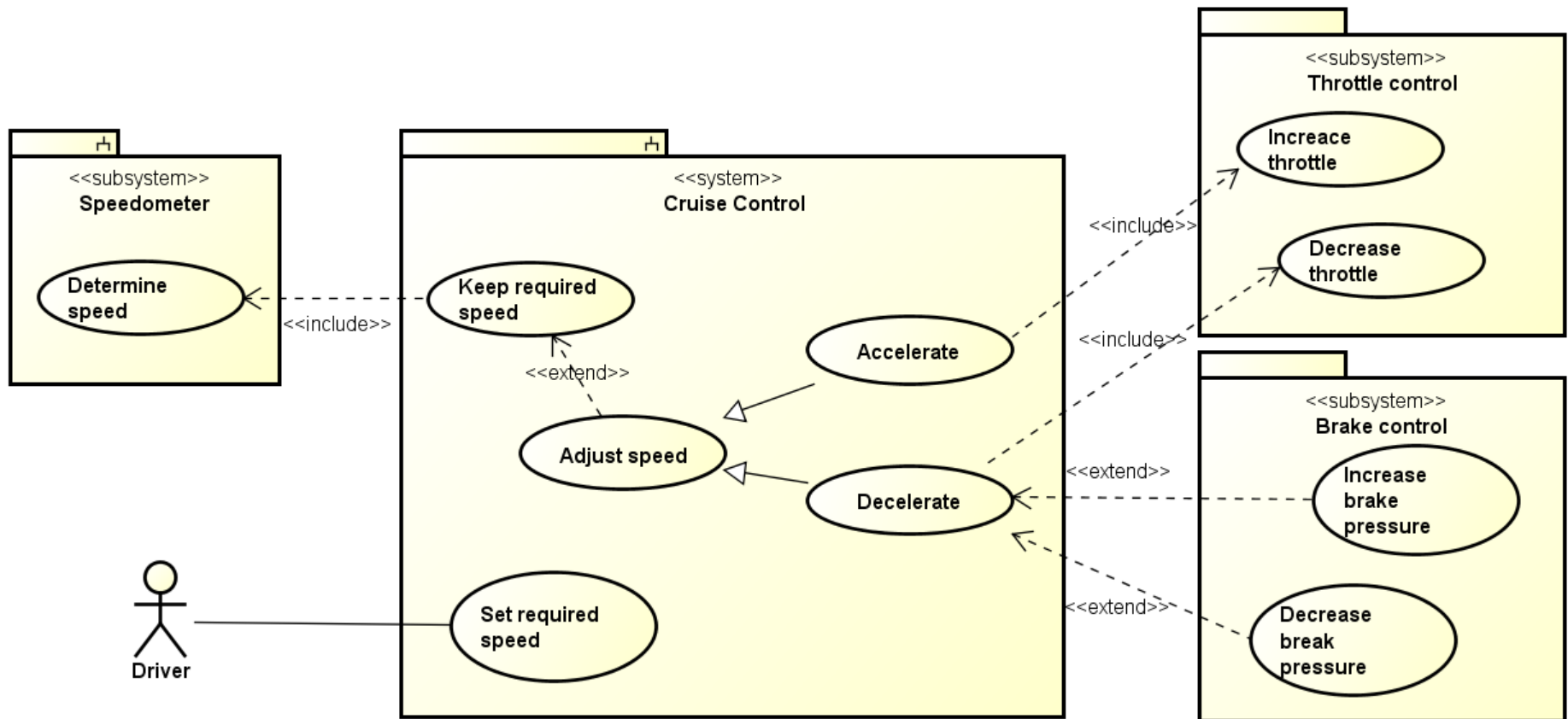
Welke subsystemen spelen hier waarschijnlijk een rol?

Extra vragen:

- Heeft de “Keep required speed” use case een actor?
- Maakt dit uit?



# Opdelen in subsystemen

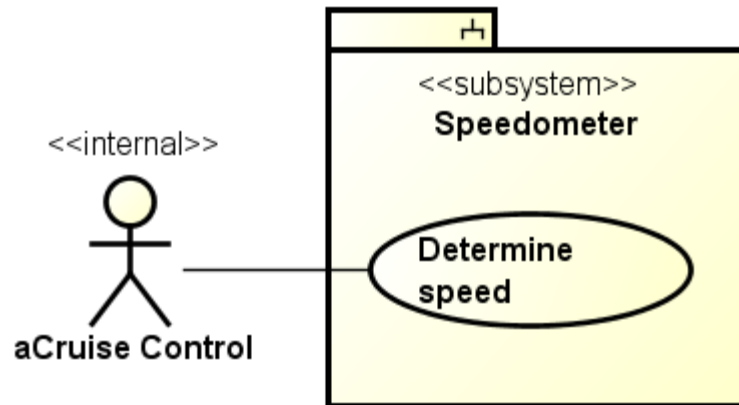


# Opdelen in subsystemen

## Subsysteem als actor

Het subsysteem fungeert hier als actor.

Het **<<internal>>** stereotype wordt hier voor gebruikt.



Opmerking:

Indien in Astah het <<internal>> stereotype wordt toegekend aan een actor dan verandert de weergave.

Hier is het als losse tekst ingevoegd.

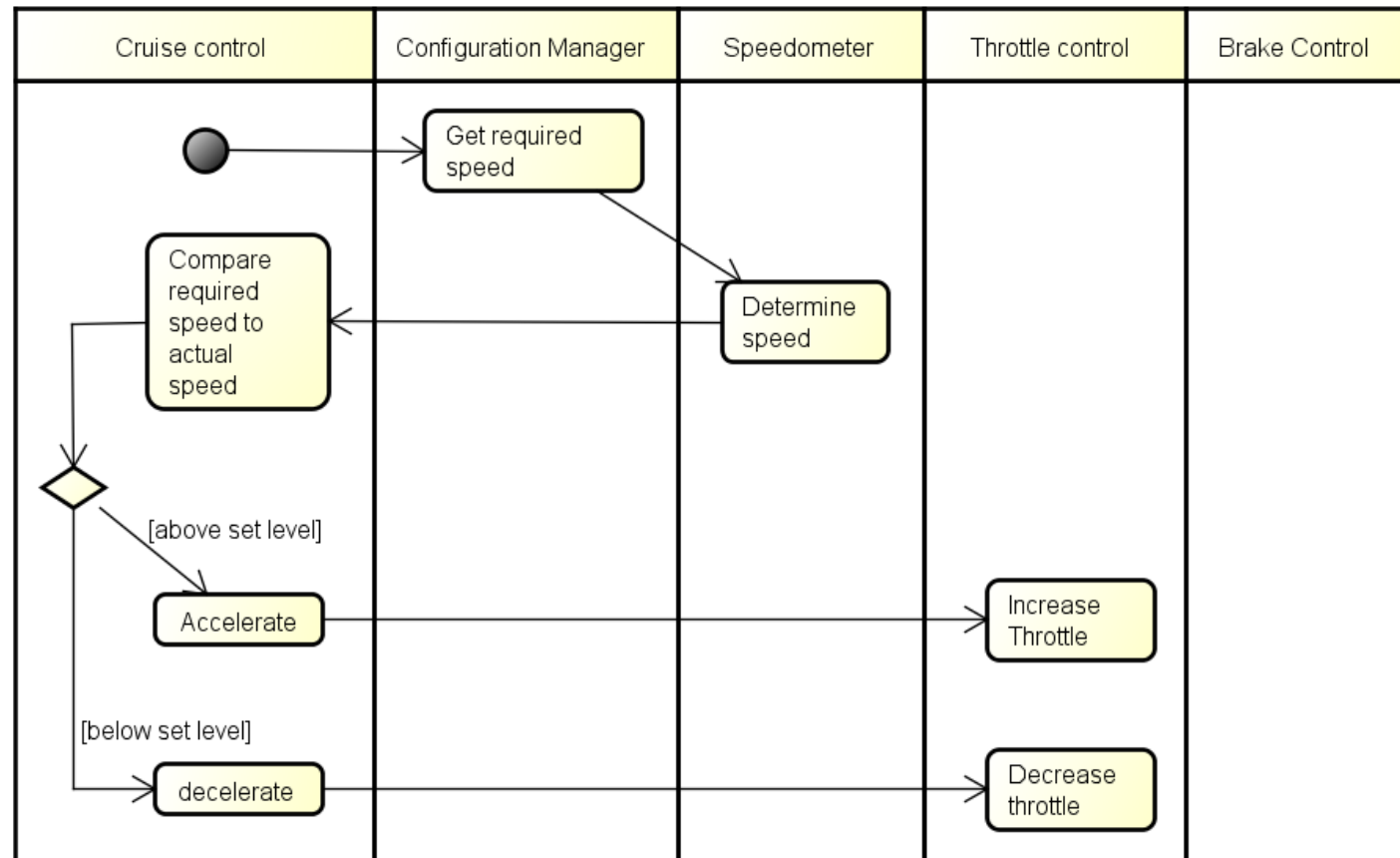
# Bepalen Use Cases (m.b.v. activity diagram)

Koppel activiteiten aan subsystemen in swimlanes

Hiermee detecteer je (missende) use cases.

Hier: “*Get required speed*” en mogelijk subsysteem.

Wat mist er nog meer?



# Opdracht (uitbreiding cruise control)

Het handhaven van de snelheid d.m.v. cruise control is handig, maar er is meer mogelijk. Daarom de volgende opdracht:

- Laat het systeem automatisch afremmen om botsingen met voorop rijdende auto's te voorkomen.

Opzet van de uitwerking:

1. Bepaal eventuele nieuwe **use cases** en **actoren** van het **hoofdsysteem**.
2. Bepaal nieuwe **deelsystemen** en bijbehorende **use cases**:
  - Doe dit op basis van een **activity diagram**.
  - Aanpassen van bestaande deelsystemen is toegestaan.
3. Stel het **use case diagram** op van het belangrijkste nieuwe subsysteem.

# Nabespreken (cruise control)

- Ga bij een ander groepje zitten en bespreek beide uitwerkingen.
- Richt je hierbij op het volgende:
  - Correctheid use-case en actoren hoofdsysteem
  - Substelsysteem-indeling en hierbij gemaakte keuzes.