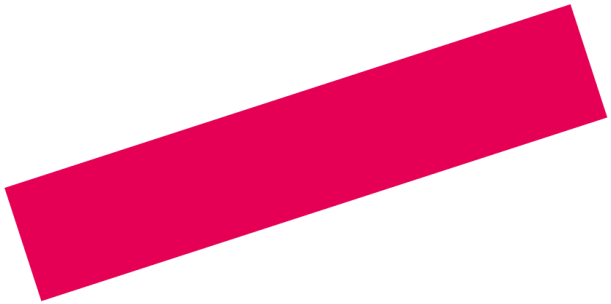
HBO-ICT Embedded Software Development

Hardware-simulatie



World of Robots :: World

Lesdoelen

- Het belang van testen met stubs kunnen benoemen
- Het kunnen bepalen van minimumeigenschappen van interfaces waar stubs aan moeten voldoen
- Het kunnen implementeren van een eenvoudige stub voor een hardwarecomponent in ROS

Verschillende tests

Unit testing

Testen van de kleinst mogelijke eenheden in een systeem in isolatie

Integration testing

Testen van interacties tussen componenten

System testing

Testen van volledige usecases

Test helpers

Testdriver(s)

Code die zorgt dat de juiste objecten er zijn en de juiste methoden aangeroepen worden om de te testen state te bereiken

Stub (ook wel *mock* of *fake*)

"Nep"-object dat wordt gebruikt door het System Under Test (SUT). Meestal voorzien van hard-coded reacties.

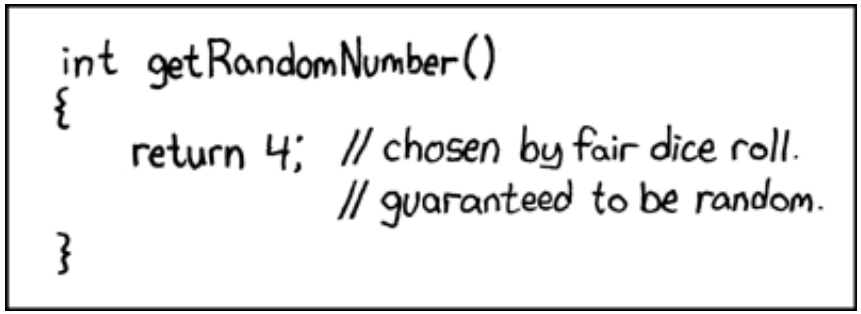
Method stub – uiteindelijke implementatie

```
char Dice::roll()
{
  static std::random_device rd;
  static std::mt19937 gen(rd());
  static std::uniform_int_distribution<> dis(1, 6);
  value = dis(gen);
  return value;
}
```

Wat zou een mogelijke stub-implementatie zijn?

Method stub – Mogelijke stub implementatie

```
char Dice::roll()
{
  return 2;
}
```



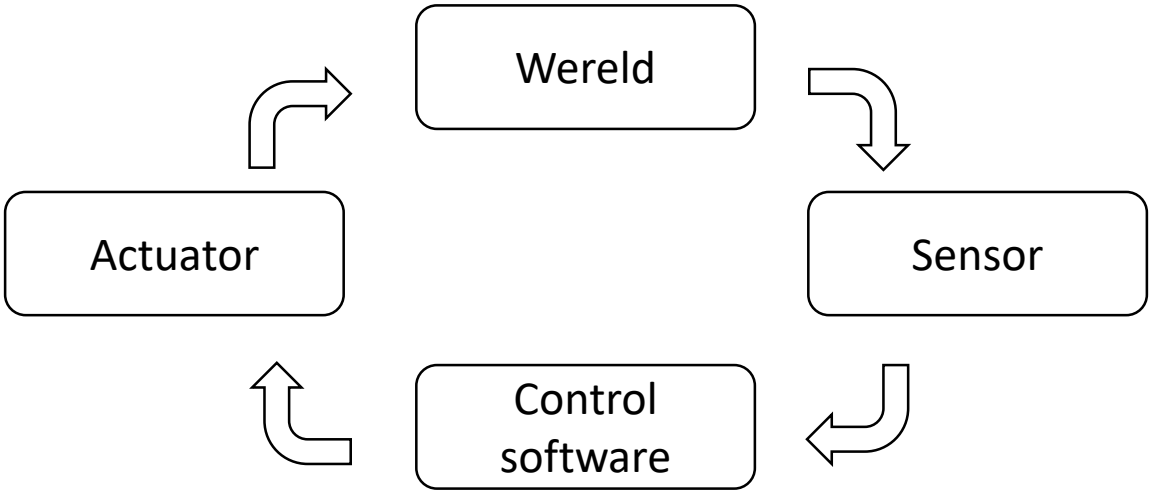
https://imgs.xkcd.com/comics/random_number.png

Wat moet ik hiermee?

- Bij Test Driven Development (TDD) worden stubs veel ingezet.
  - Zo kunnen unit-tests al vroeg worden geschreven.

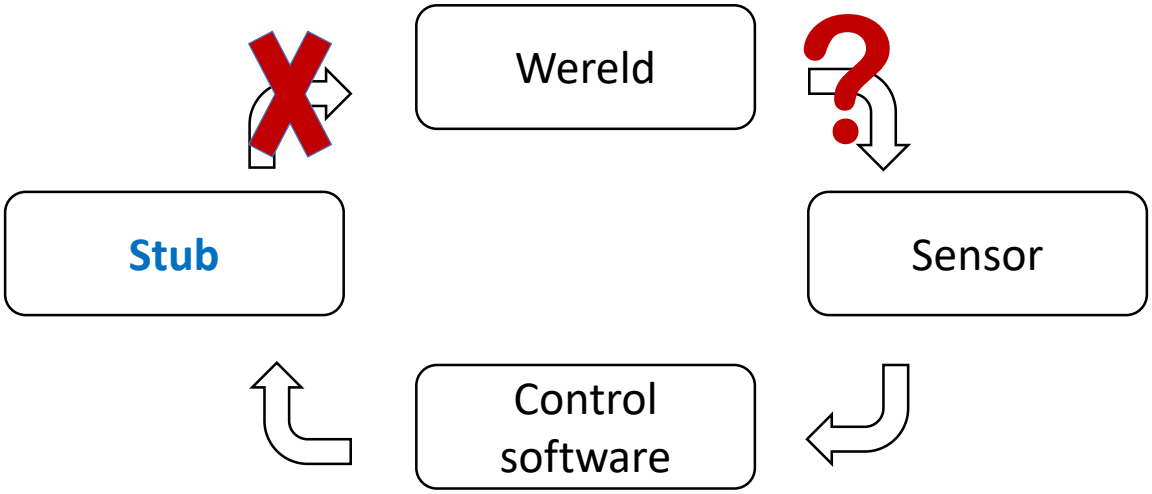
- In het ESD-vakgebied wordt vaak gewerkt met stubs voor op dat moment nog niet beschikbare hardware.

Als hardware ontbreekt…



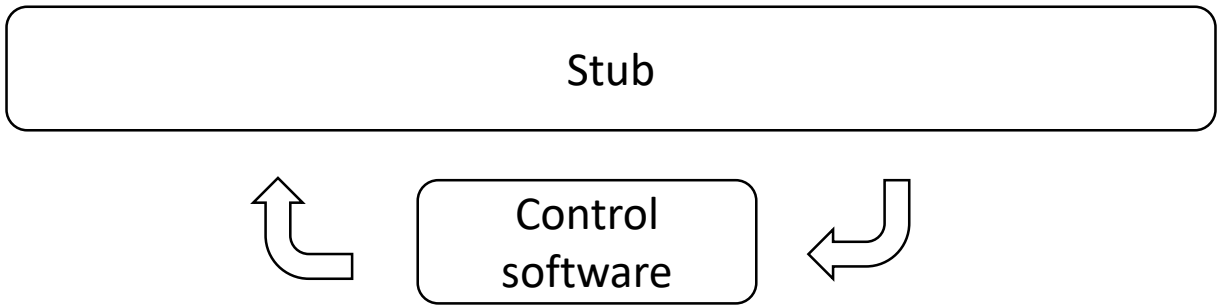
Actuator beïnvloed *indirect* ook sensoren!

Als hardware ontbreekt…



Actuator beïnvloed *indirect* ook sensoren!

Als hardware ontbreekt…



Een stub moet daarom vaak naast een actuator ook sensoren vervangen.

Praktisch – In termen van deze course

Een stub is een component met dezelfde interfaces als het uiteindelijke te gebruiken component, waarbij:

- wel wordt voldaan aan de volledige interface-specificatie en pre-condities.
- niet per se wordt voldaan aan:
  - hetzelfde behavior, of
  - de postcondities.

Wat zijn de nadelen van zo'n stub?

Praktisch – In termen van deze course

Een stub is een component met dezelfde interfaces als het uiteindelijke te gebruiken component, waarbij:

- wel wordt voldaan aan de volledige interface-specificatie en pre-condities.
- niet per se wordt voldaan aan:
  - hetzelfde behavior, of
  - de postcondities.

Nadelen van zo'n stub:

- Beperkte inzetbaarheid als correct gedrag en resultaten belangrijk zijn voor juist systeemgedrag.
  Zo zullen uitgebreide integratietesten niet slagen.

Praktisch – Reduceren van stub-nadelen

- De simpelste stub-implementatie retourneert slechts valide waarden.

- De meest uitgebreide betreft een volledige simulatie.

- Meestal is er een bruikbare tussenweg:
  Een stub kan anders reageren op basis van:
  - volgorde van aanroepen,
  - verstreken tijd,
  - gegeven argumenten,
  - et cetera.

Simulatieopdracht – Beker-stub

Voordat we een zo realistisch mogelijke beker-simulatie realiseren maken we eerst een stub.

- Zo kunnen we al vroeg beginnen met systeemintegratie en testen.

Schrijf een stub ROS node voor een beker:

- Wat is de benodigde interface?
- Wat is minimale benodigde test-output?