



Vattenfall Solar Team

Testing software for Nuna



Pieter Tolsma

17-05-2019

Quick introduction



Australia



Vattenfall Solar Team

Clear goal



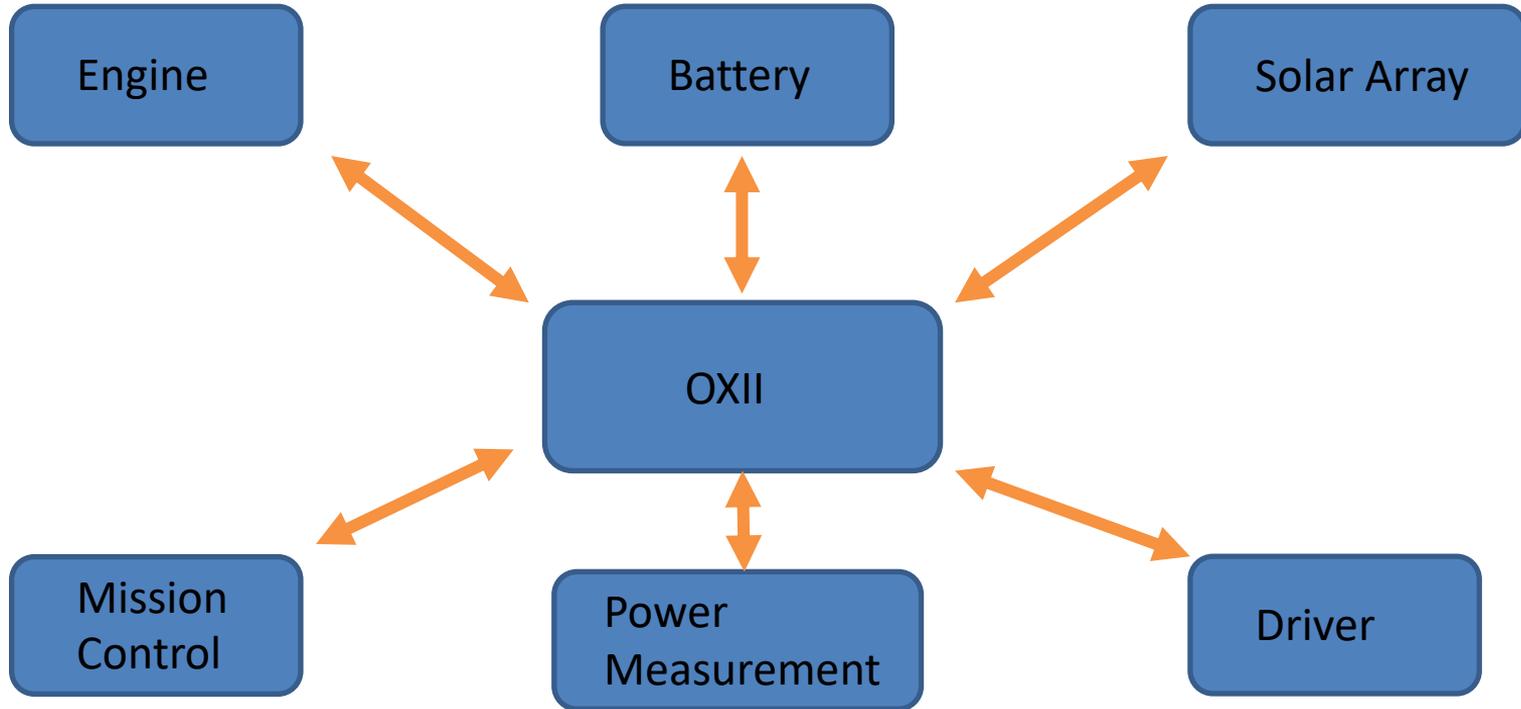
My role in the team

Testdag
verbruiksmodel
dracien

Testdag 11
voor verbruiks-
model
worden



OXII, the on-board computer



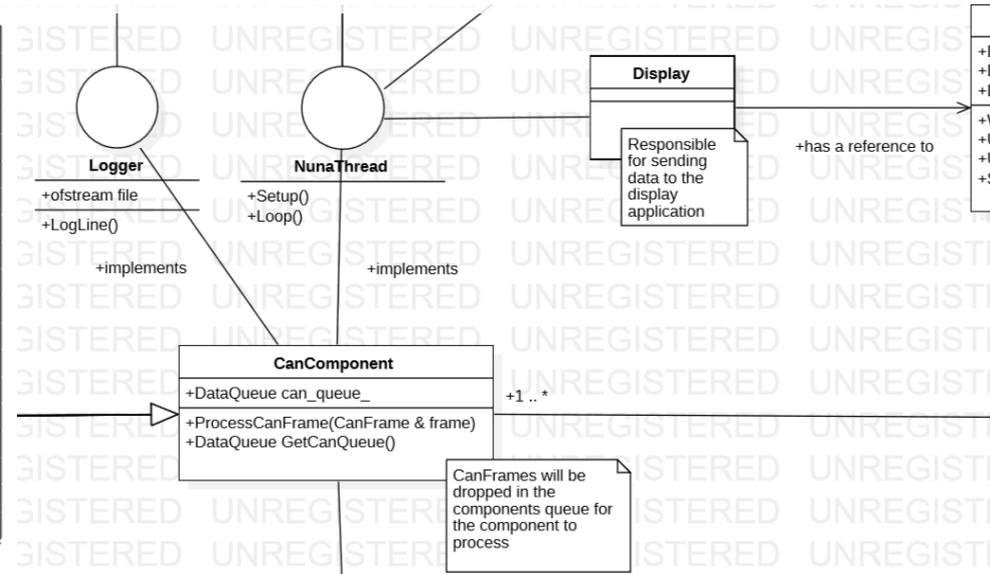


Lessons learned



Good system analysis is more important than writing code

CAN ID	Component	Description	Rate in	Rate out
0x010	Steer	Steer message	200ms	
0x0B	Motor	Tritium motor temp message	1s	
0x0C	Motor	Tritium board temp message	1s	
0x200	CMS	Plus Minus Box	200ms	
0x201	CMS	Battery Current	200ms	
0x202 + n	MPPT	Power throttling command		200ms
0x220 + n	MPPT	Input current / voltage MPPT	200ms	
0x230 + n	MPPT	Input power and output voltage	200ms	
0x250 + n	MPPT	Temperature of MPPT	200ms	
0x300 + m	Battery	REAP 21 Status	200ms	
0x501	Motor	Tritium velocity command message		100ms
0x502	Motor	Tritium power command message		100ms
0x580 + m	Battery	REAP 21 Values of interest	200ms	
0x630 + m	Battery	REAP 21 Summary	200ms	
0x7C0	Motor	Tritium ID message	1s	
0x7C1	Motor	Tritium status message	200ms	
0x7C3	Motor	Tritium velocity message	200ms	
0x7C0	Motor	Tritium ID message	1s	



Importance of reliability



Embrace unpredictability

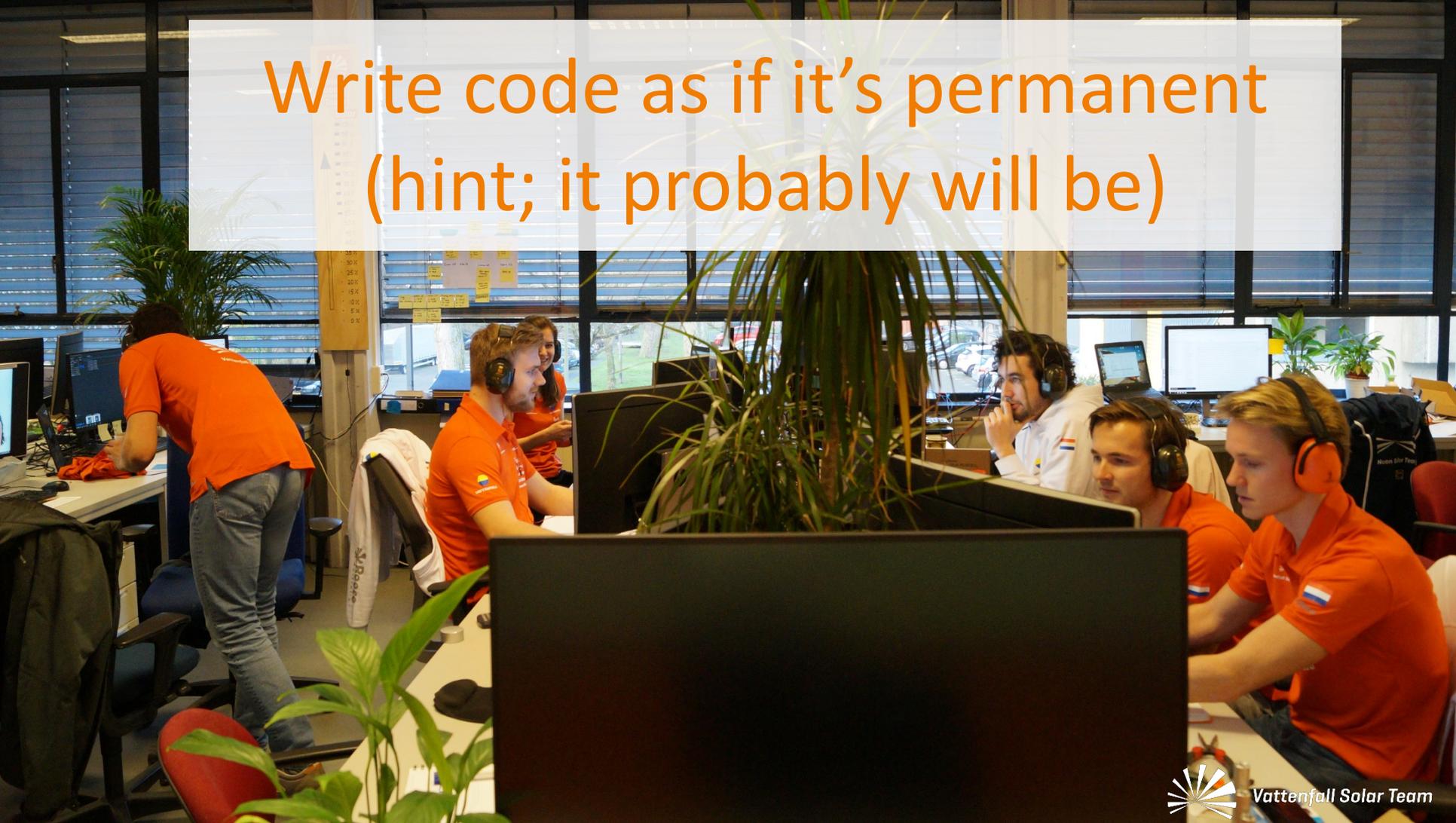


Communication should be
your number 1 priority.



Vattenfall Solar Team

Write code as if it's permanent
(hint; it probably will be)



What I learned and applied from this course

- Deep requirements analysis.
- Keep it simple, stupid; code should be easy to read.
- **Write modularized, testable code.**
- Documentation.
- Solid workflow.
- Steady releases.

Example of isolated code

```
37 MissionControlSensor mission_control_sensor("Mission Control Sensor", ethernet_buffer);
38 Nuna9BatteryPack n9_pack("Nuna9 Battery", can_buffer0, cms_sensor, mission_control_sensor);
39 BatterySensor &battery_sensor(n9_pack);
40 XSensor xsensor("XSensor", xsens_buffer);
41 RadarSensor radar_sensor("Radar Sensor", can_buffer1, xsensor, motor_sensor, steer_sensor);
42 MPPTSensor mppt_sensor("MPPT Sensor", can_buffer0, n9_pack);
43 TargetSpeedSensor target_speed_sensor("TargetSpeed Sensor", ethernet_buffer, motor_sensor, mission_control_sensor);
```

44

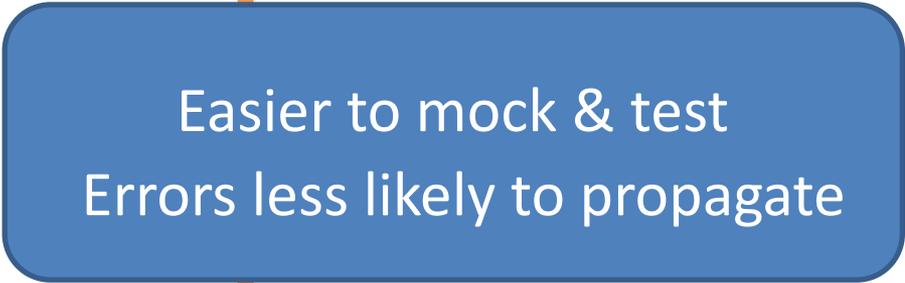


```
80 TritiumController controller(nuna);
81
82 Motor motor(nuna, controller);
83
84 MPPT mppt(nuna);
85
86 CMS cms(nuna);
87 Display display(nuna);
88
89 MissionControl mission_control(nuna);
```

Example of isolated code

```
37 MissionControlSensor mission_control_sensor("Mission Control Sensor", ethernet_buffer);
38 Nuna9BatteryPack n9_pack("Nuna9 Battery", can_buffer0, cms_sensor, mission_control_sensor);
39 BatterySensor &battery_sensor(n9_pack);
40 XSensor xsensor("XSensor", xsens_buffer);
41 RadarSensor radar_sensor("Radar Sensor", can_buffer1, xsensor, motor_sensor, steer_sensor);
42 MPPTSensor mppt_sensor("MPPT Sensor", can_buffer0, n9_pack);
43 TargetSpeedSensor target_speed_sensor("TargetSpeed Sensor", ethernet_buffer, motor_sensor, mission_control_sensor);
```

44



Easier to mock & test
Errors less likely to propagate

```
80 TritiumController controller(nuna);
81
82 Motor motor(nuna, controller);
83
84 MPPT mppt(nuna);
85
86 CMS cms(nuna);
87 Display display(nuna);
88
89 MissionControl mission_control(nuna);
```

Realtime systems are ...

- Unpredictable
- Hard to test
- About milliseconds.
- ... so how do we test it?

Meet Benny



Our system testing



Even better system testing



SIG code review

Overview of the maintainability rating

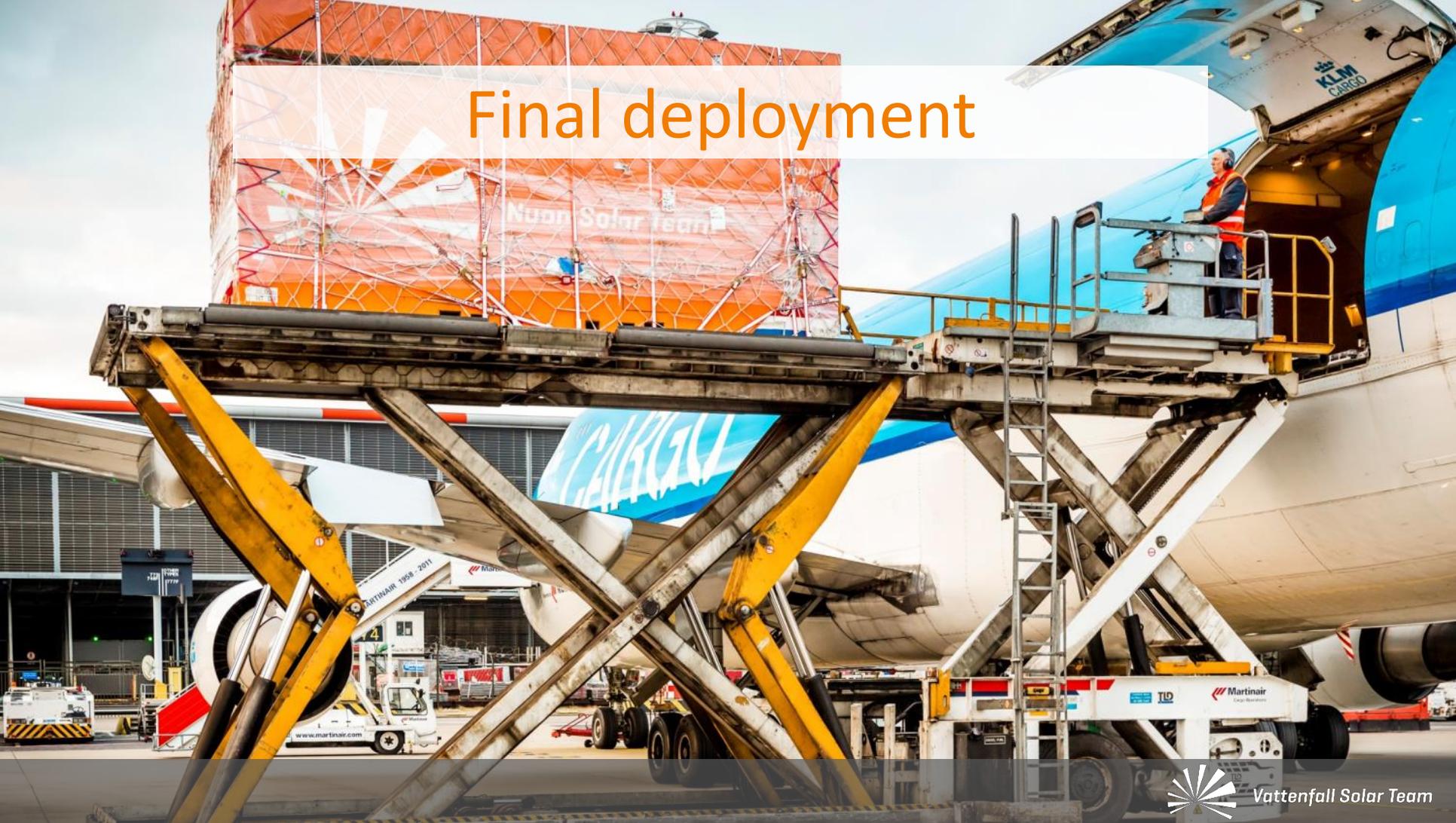
VattenfallSolarTeam - Oxii (3.9)



Volume	★★★★★	5.5
Duplication	★★★★★	4.6
Unit size	★★★★☆	3.9
Unit complexity	★★★★☆	3.8
Unit interfacing	★★★★☆	4.4
Module coupling	★★★☆☆	2.8
Component balance	★★★★★	4.6
Component independence	★★☆☆☆	1.9



Final deployment



Race! (The ultimate test)



Race!



Vattenfall Solar Team



Vattenfall Solar Team

Testing software for Nuna



Pieter Tolsma

17-05-2019