

University of Stuttgart  
Germany

caps lock

safety


# Using requirements specification to speed up STPA-BDD in agile development

Amsterdam, 2018-11-02

**Stefan  
Wagner**





# You can

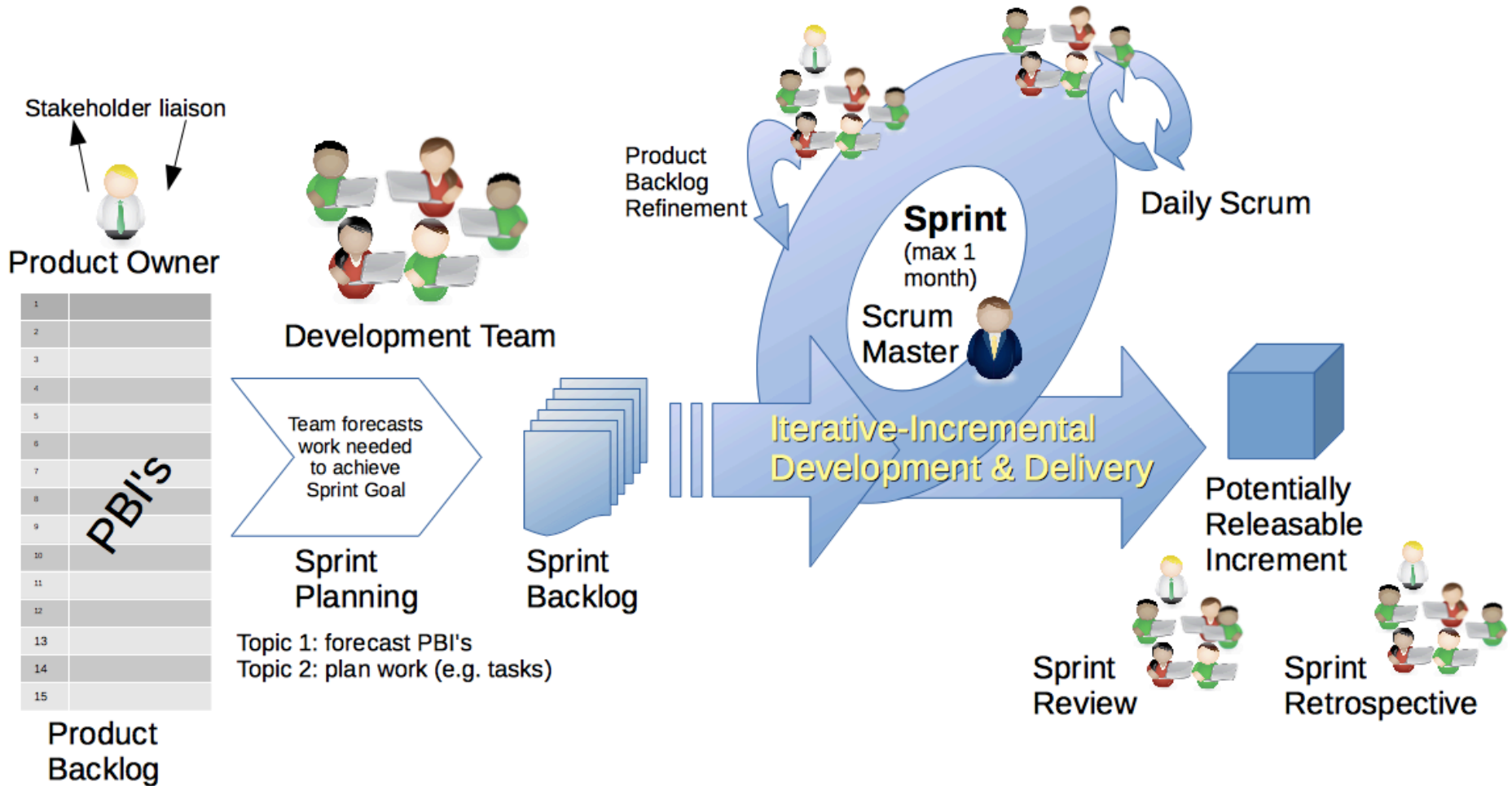
  copy, share and change,

  film and photograph,

  blog, live-blog and tweet

 this presentation given that you attribute  
 it to its author and respect the rights and  
licences of its parts.

# Agile Software Development

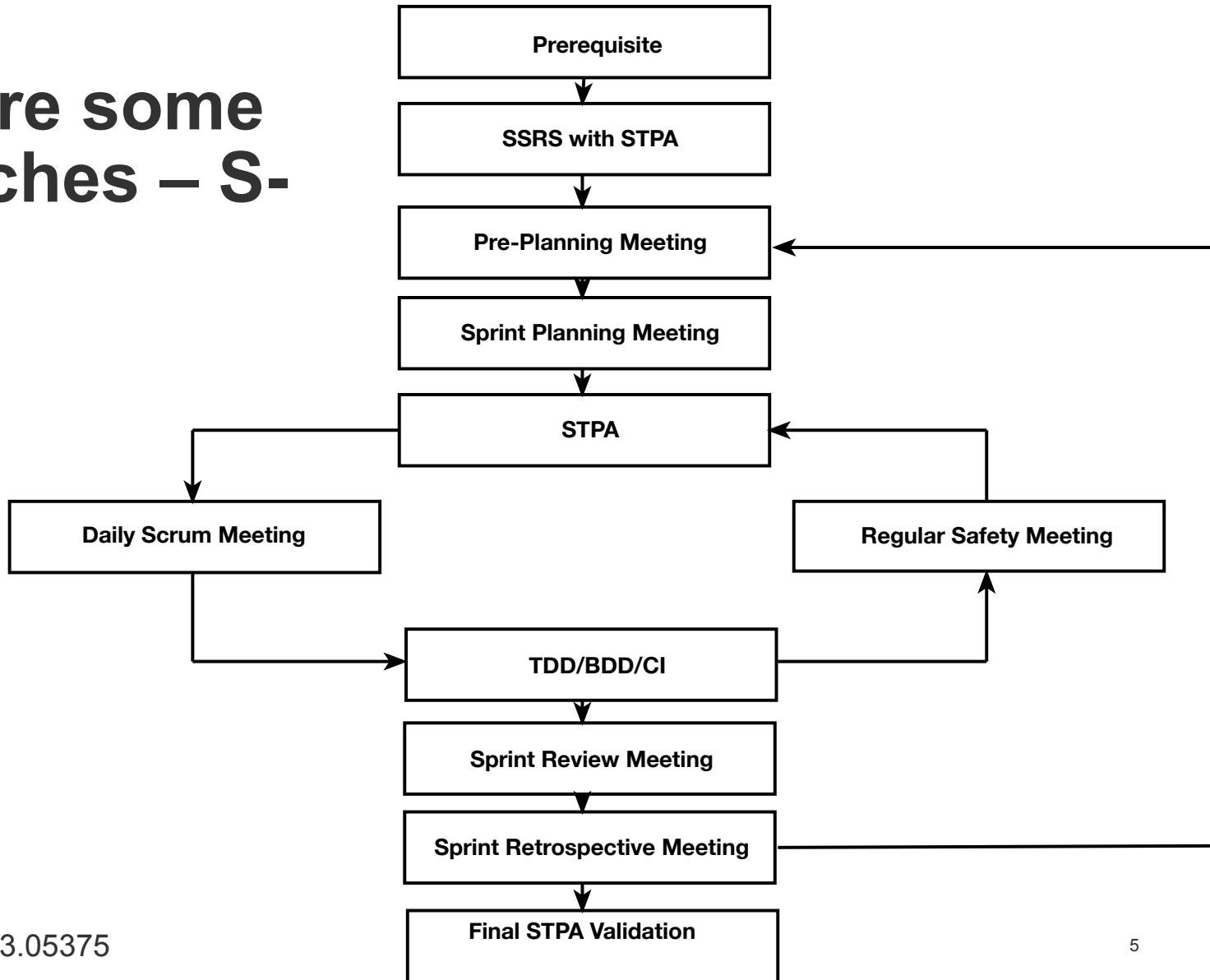


# Agile Software Development of Safety-Critical Systems?

Safety analysis without an upfront architecture design?

Unstable requirements that can change every few weeks?

# There are some approaches – S-Scrum



# Manifesto for Agile Software Development

We are uncovering better ways of  
software by doing it and helping  
Through this work we have come to value:

Focus on  
communication

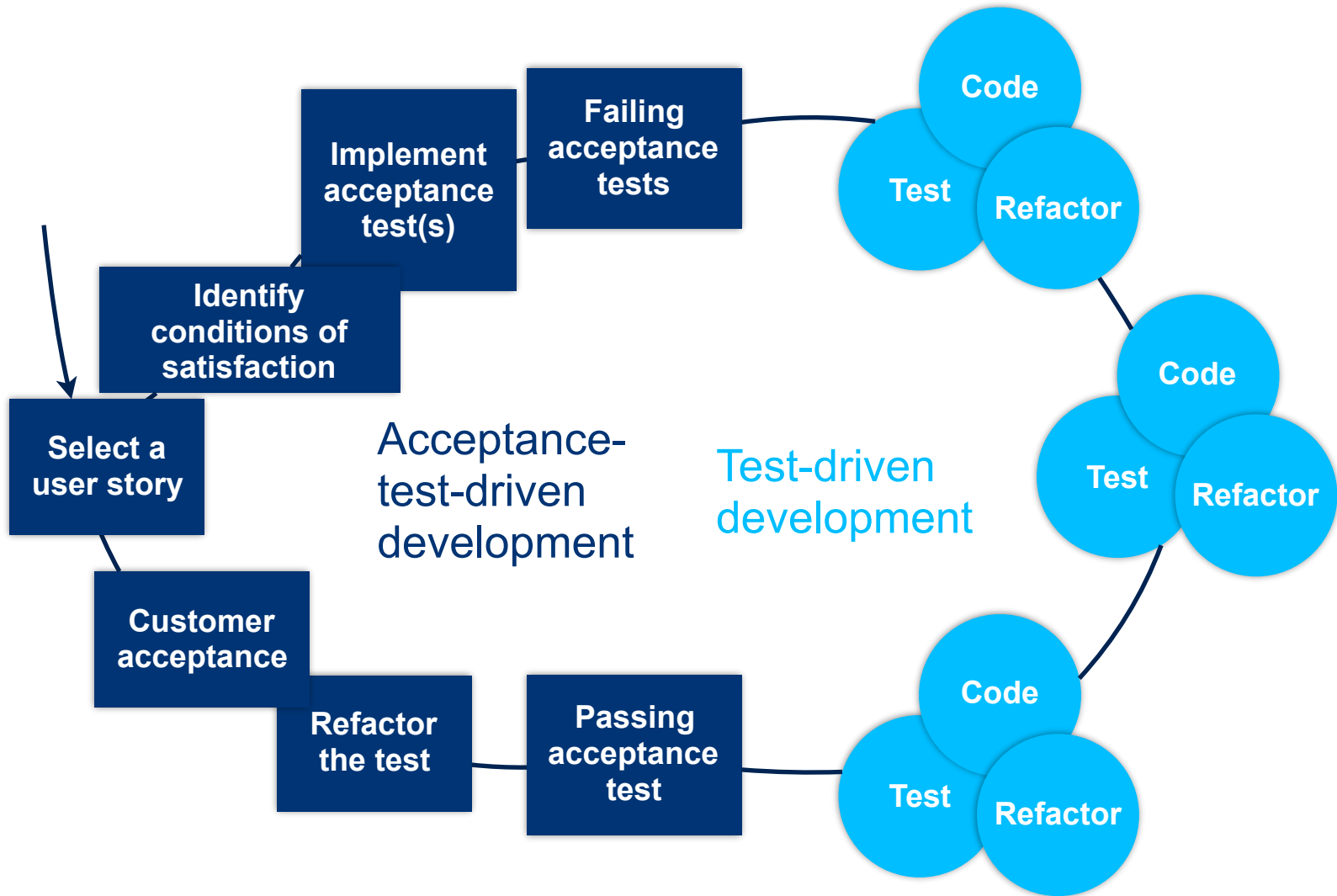
**Individuals and interactions** over processes and tools

**Working software** over comprehensive documentation

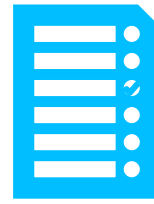
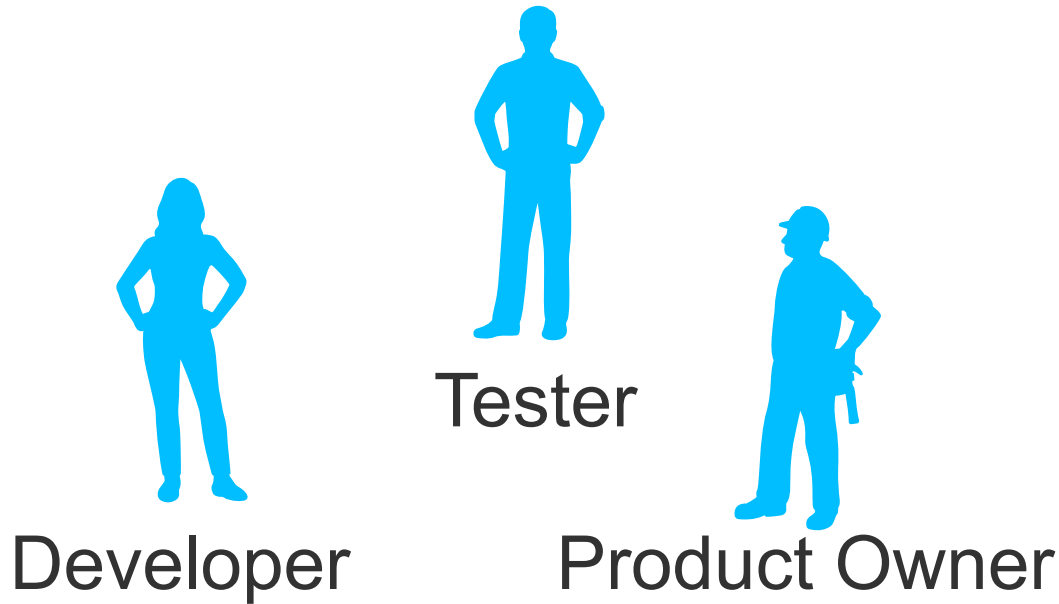
**Customer collaboration** over contract negotiation

**Responding to change** over following a plan

That is, while there is value in the items on  
the right, we value the items on the left more.



# Behaviour-Driven Development (BDD)



Examples



Scenarios



Automated Tests <sup>8</sup>



# Behaviour-Driven Development (BDD)

**Feature:** Refund item

**Scenario:** Jeff returns a faulty microwave

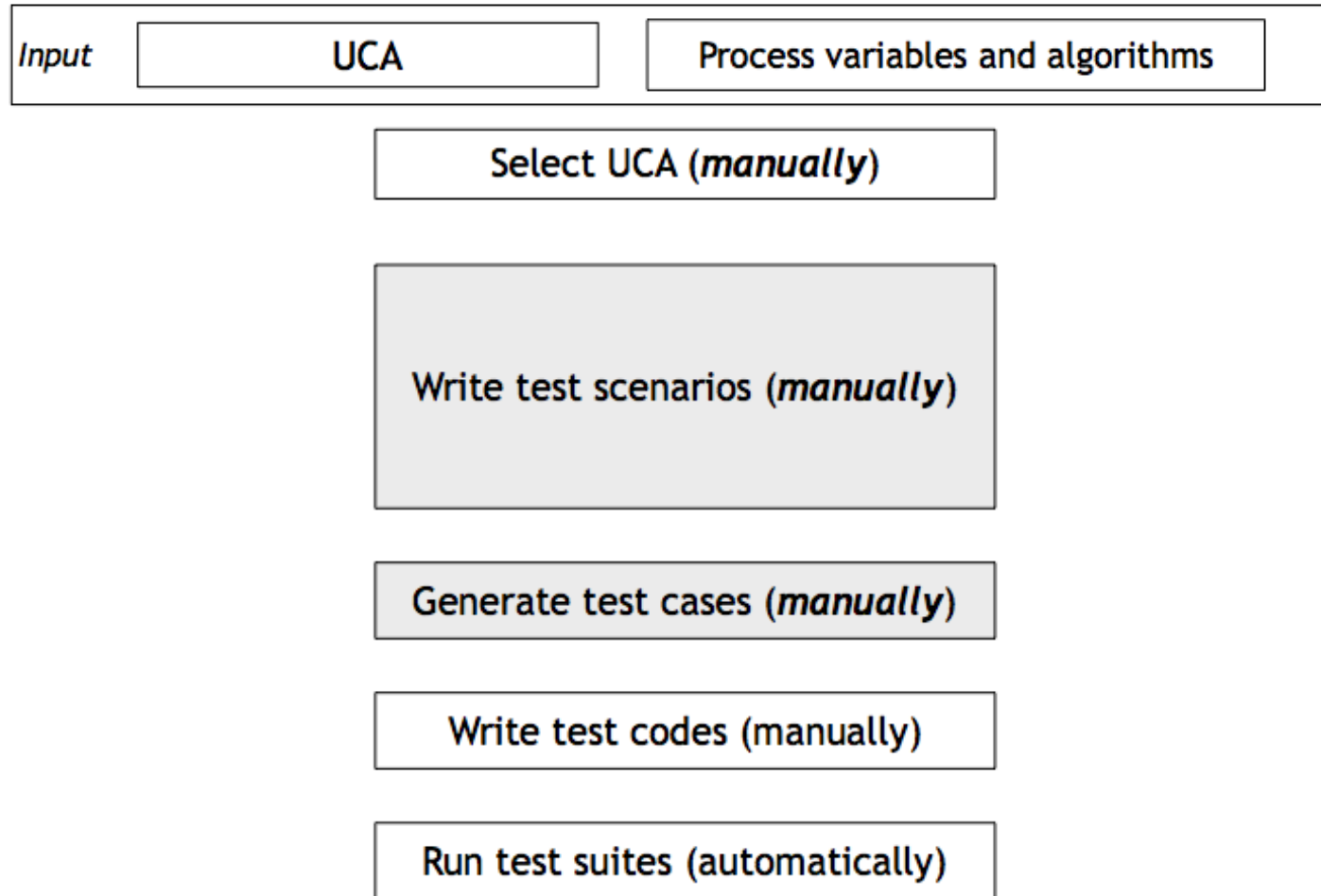
**Given** Jeff has bought a microwave for \$100

**And** he has a receipt

**When** he returns the microwave

**Then** Jeff should be refunded \$100

# STPA-BDD



# Example

## Unsafe Scenario from STPA

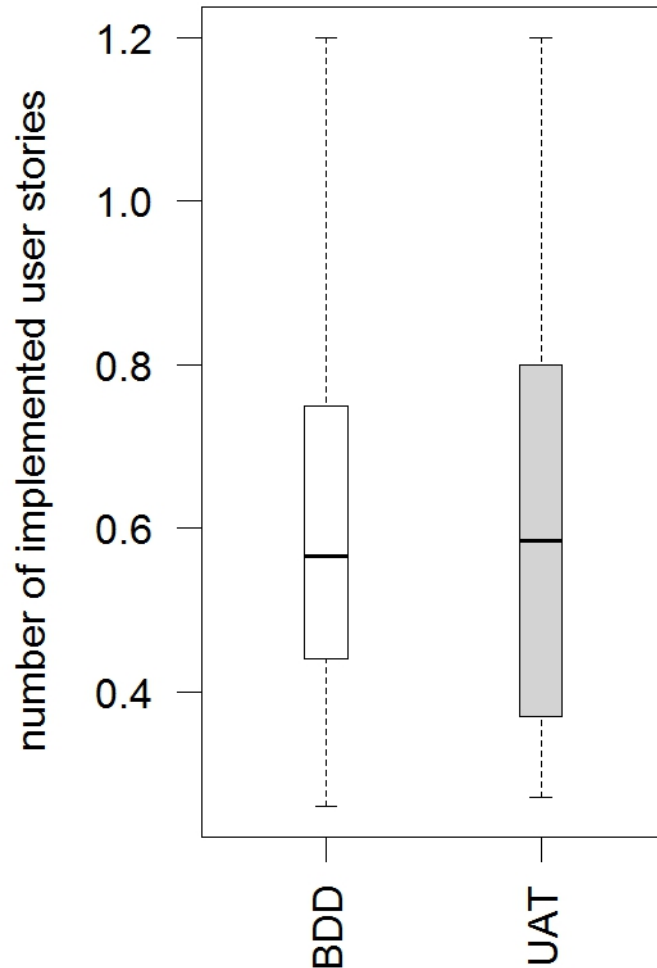
During auto-parking, the autonomous vehicle does not stop immediately when there is an obstacle up front.



## Gherkin Scenario

**Given** the autonomous vehicle is auto-parking  
**When** the ultrasonic sensor provides the feedback that the forward distance is smaller or equal to a threshold indicating that there is an obstacle up front  
**Then** the autonomous vehicle stops immediately.

## Experimental results



**But: Communication effectiveness is significantly different!**

The developers consider the safety requirements deeply and initiatively.

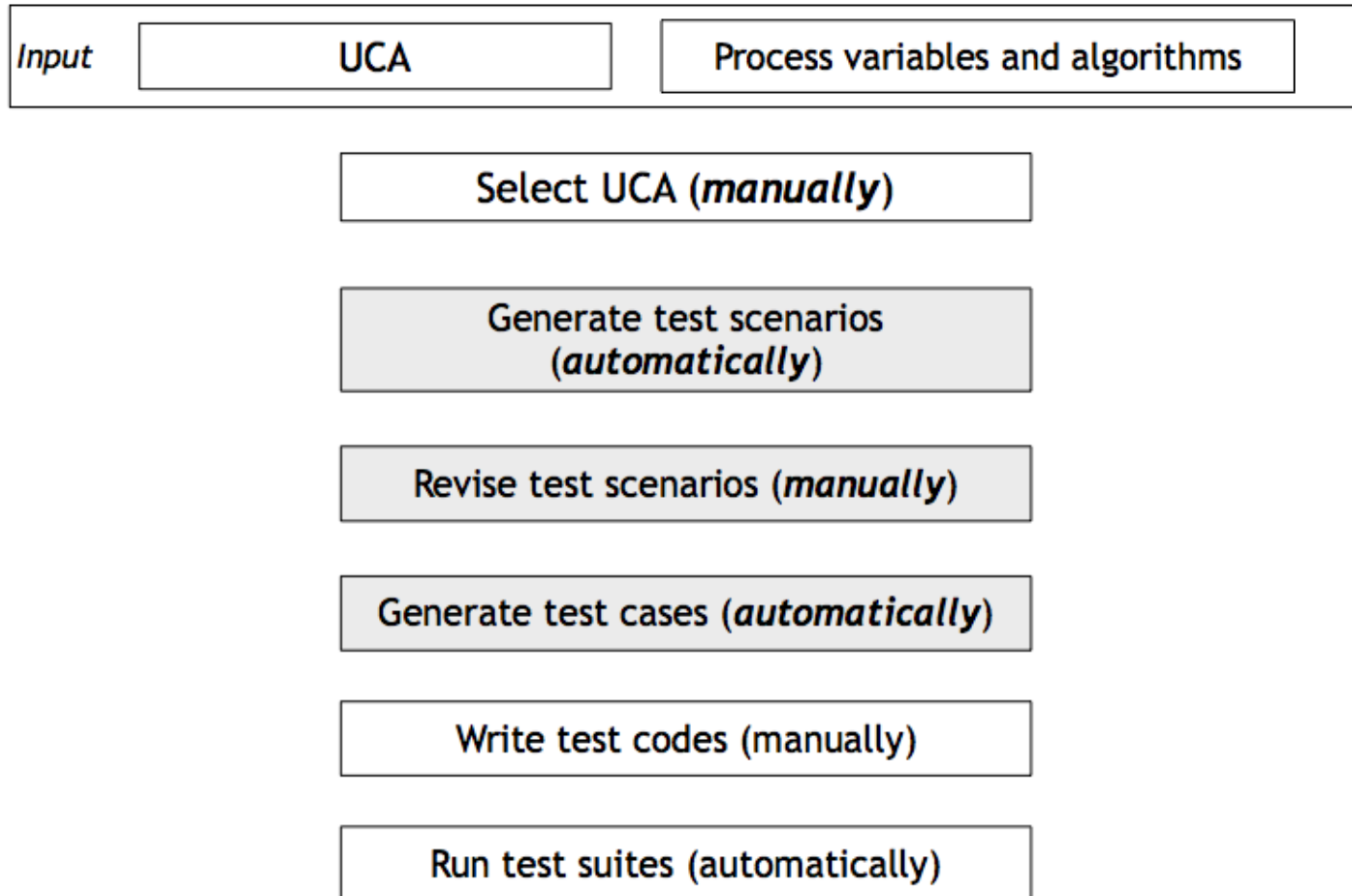
The business analysts are more confident about the test cases.

It becomes easier to identify conflicts in business rules and test cases.

The business analysts are clear about the status of acceptance testing.

The business analysts could spend less time on sprint-end acceptance tests.

# Speeding it up with automation



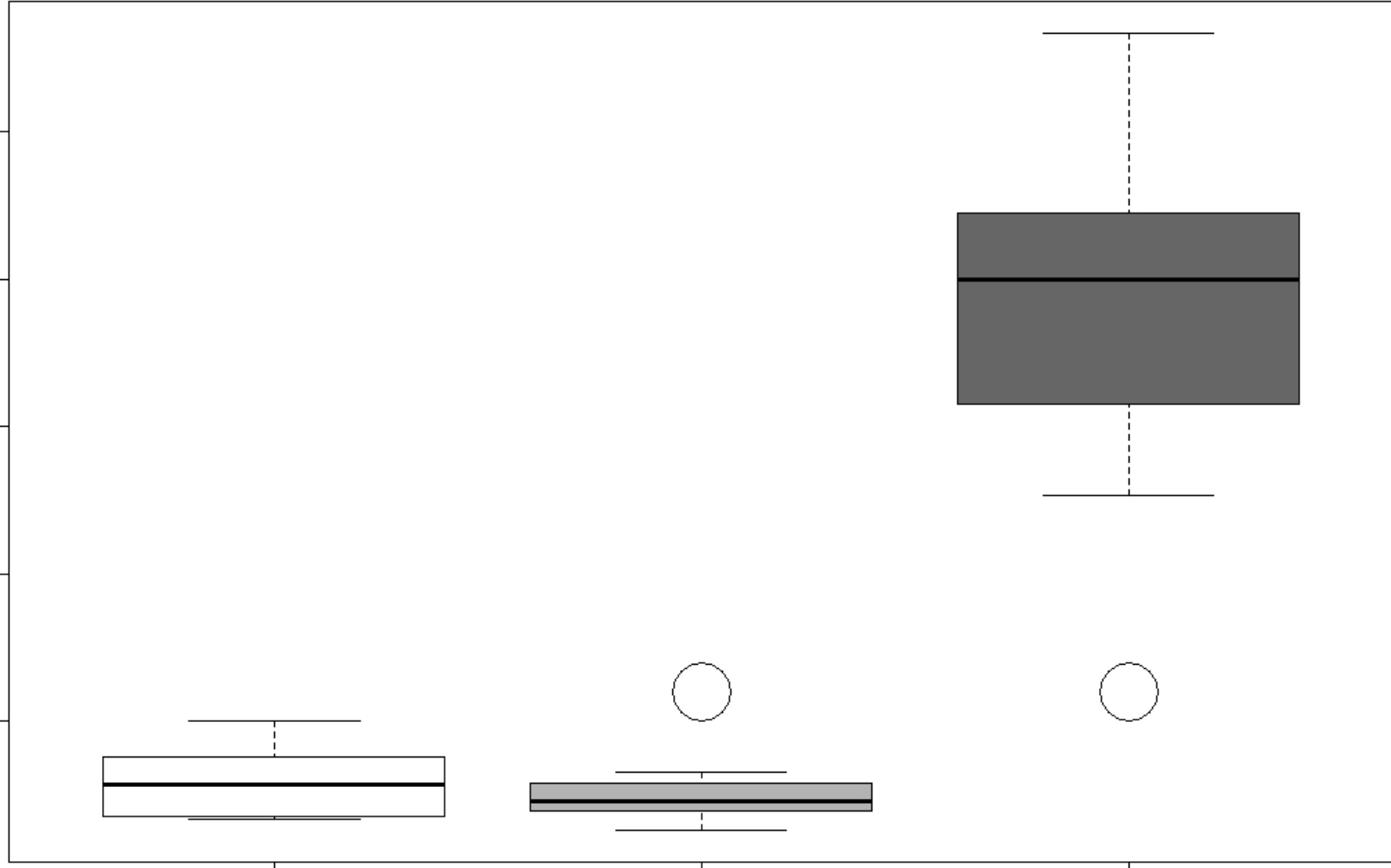
number of implemented user stories

5  
4  
3  
2  
1

UAT

BDD 1

BDD 2



# Putting a formal basis below it

*Input*

Context or unsafe scenarios

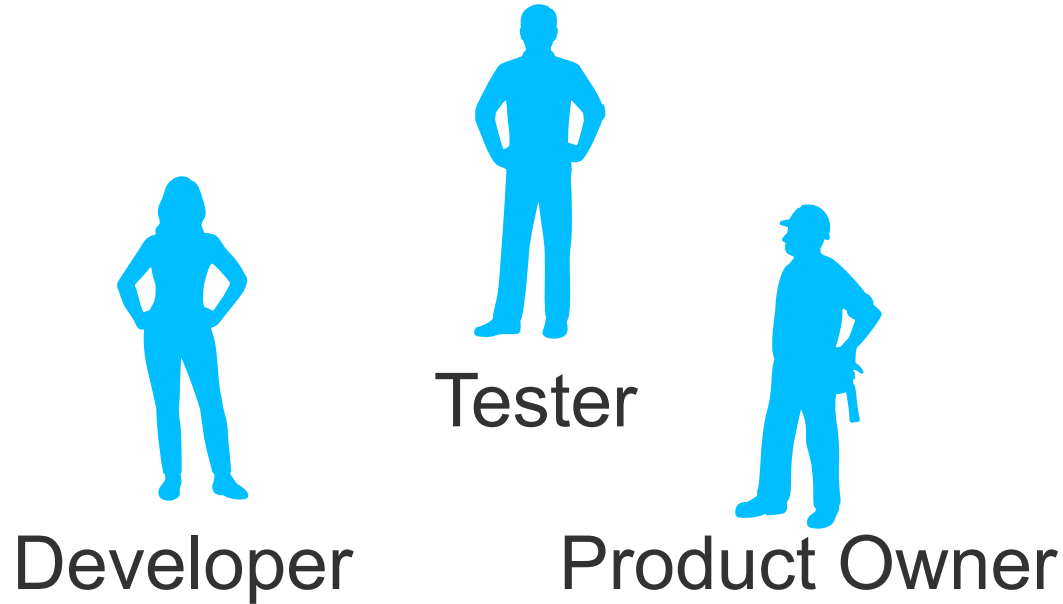
Generate test scenarios  
*(automatically)*

Generate test cases *(automatically)*

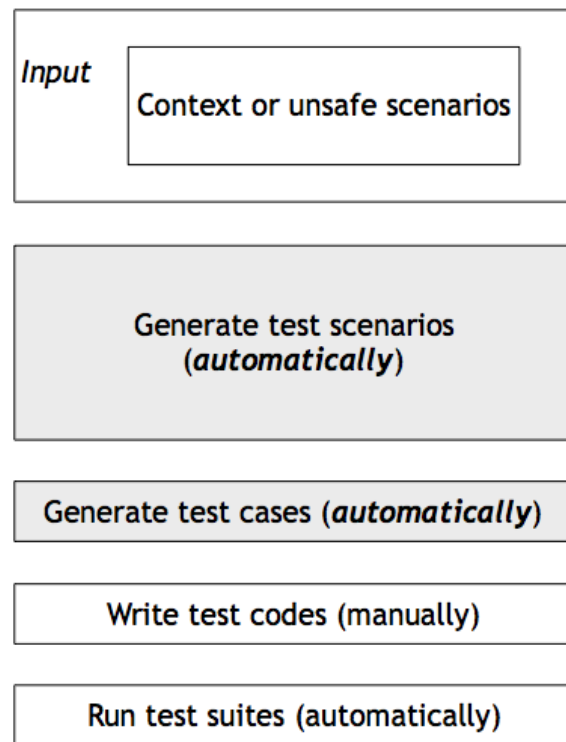
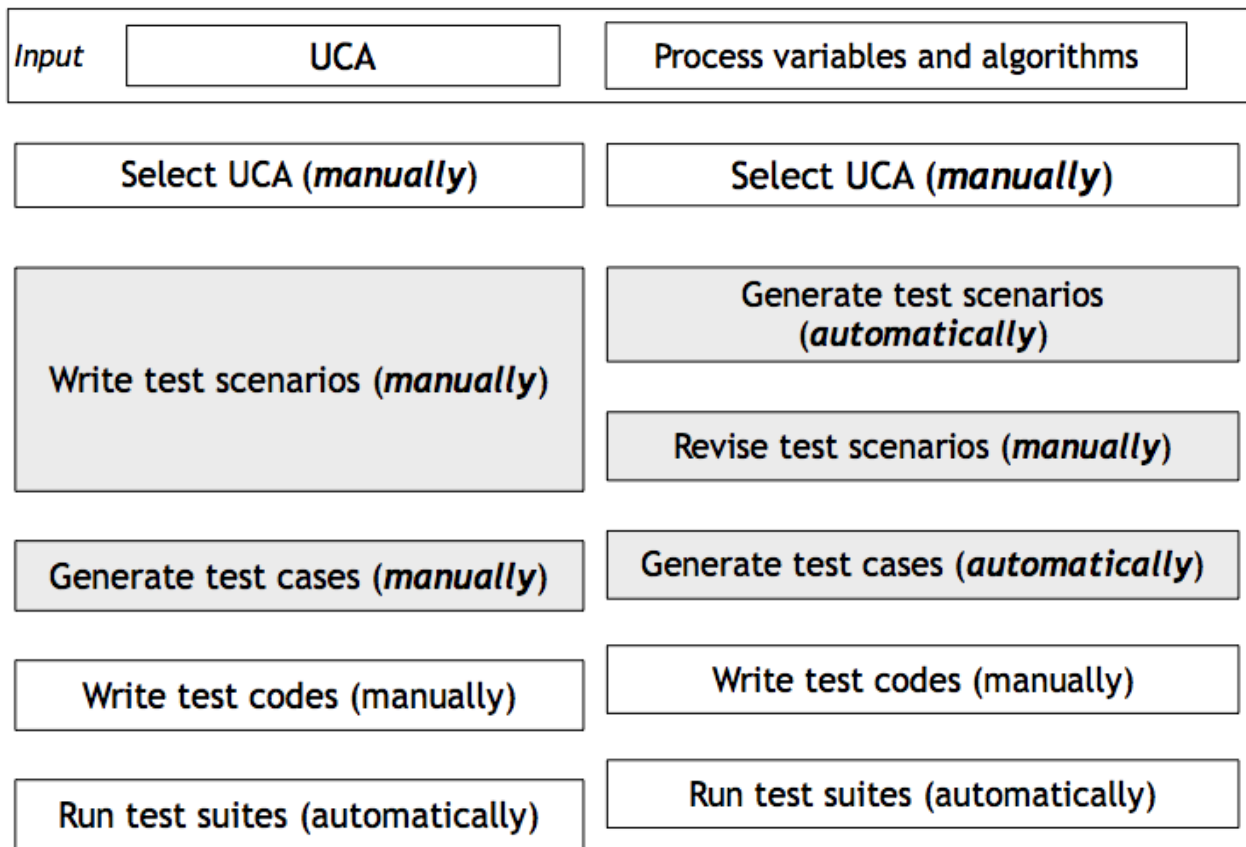
Write test codes (manually)

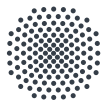
Run test suites (automatically)

# Will we loose communication?









**University of Stuttgart**  
Germany

Joint work with Yang Wang (now at Bosch)  
and John Thomas (MIT)



## **Prof. Dr. Stefan Wagner**

e-mail	<a href="mailto:stefan.wagner@informatik.uni-stuttgart.de">stefan.wagner@informatik.uni-stuttgart.de</a>
phone	+49 (0) 711 685-88455
WWW	<a href="http://www.iste.uni-stuttgart.de/se">www.iste.uni-stuttgart.de/se</a>
Twitter	<a href="https://twitter.com/prof_wagnerst">prof_wagnerst</a>
ORCID	0000-0002-5256-8429

# Pictures used in this slide deck

Safety by GotCredit (<https://flic.kr/p/qHCmfo>, [Got Credit](#))

Scrum framework by Dr ian mitchell under CC BY-SA 4.0 ([https://en.wikipedia.org/wiki/Scrum\\_\(software\\_development\)#/media/File:Scrum\\_Framework.png](https://en.wikipedia.org/wiki/Scrum_(software_development)#/media/File:Scrum_Framework.png))

Screenshot from <http://agilemanifesto.org> by Ward Cunningham