Embedded Systems
INSTITUTE

An Introduction to

Model-Based Testing

Jan Tretmans

Embedded Systems Institute, Eindhoven, NL
and Radboud University, Nijmegen, NL

(Software) Testing

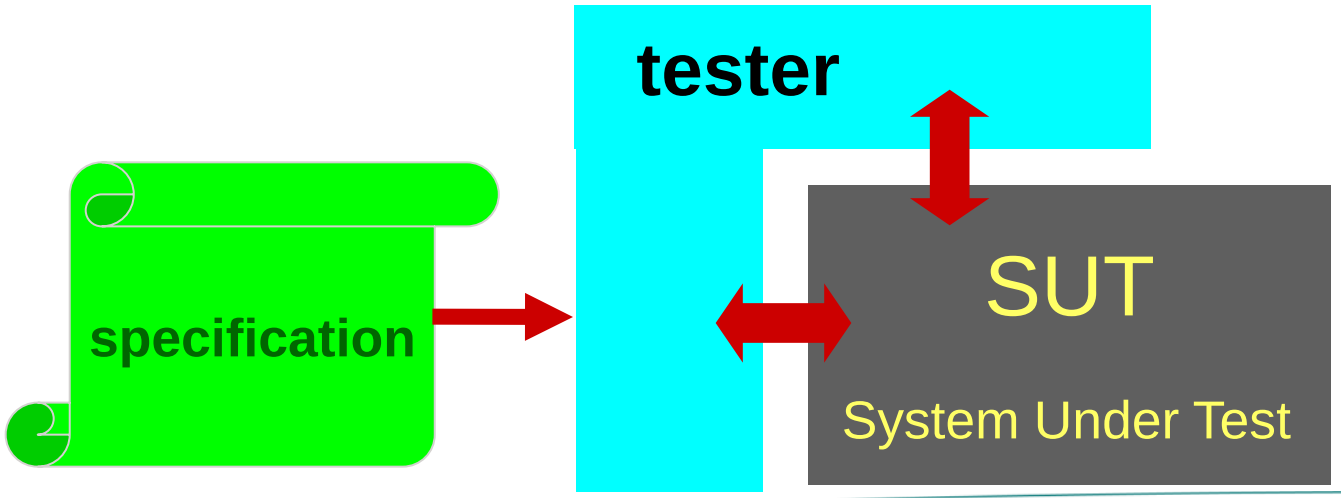
checking or measuring some quality aspects

of an executing object

by performing experiments

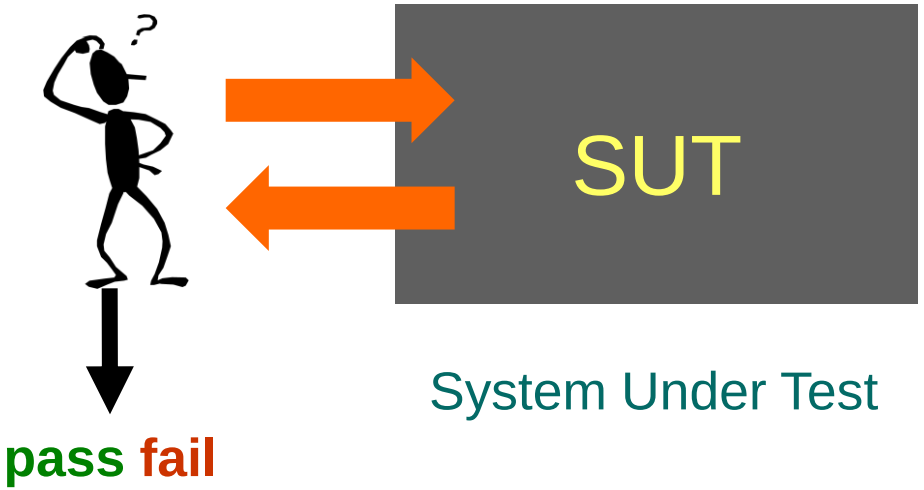
in a controlled way

w.r.t. a specification

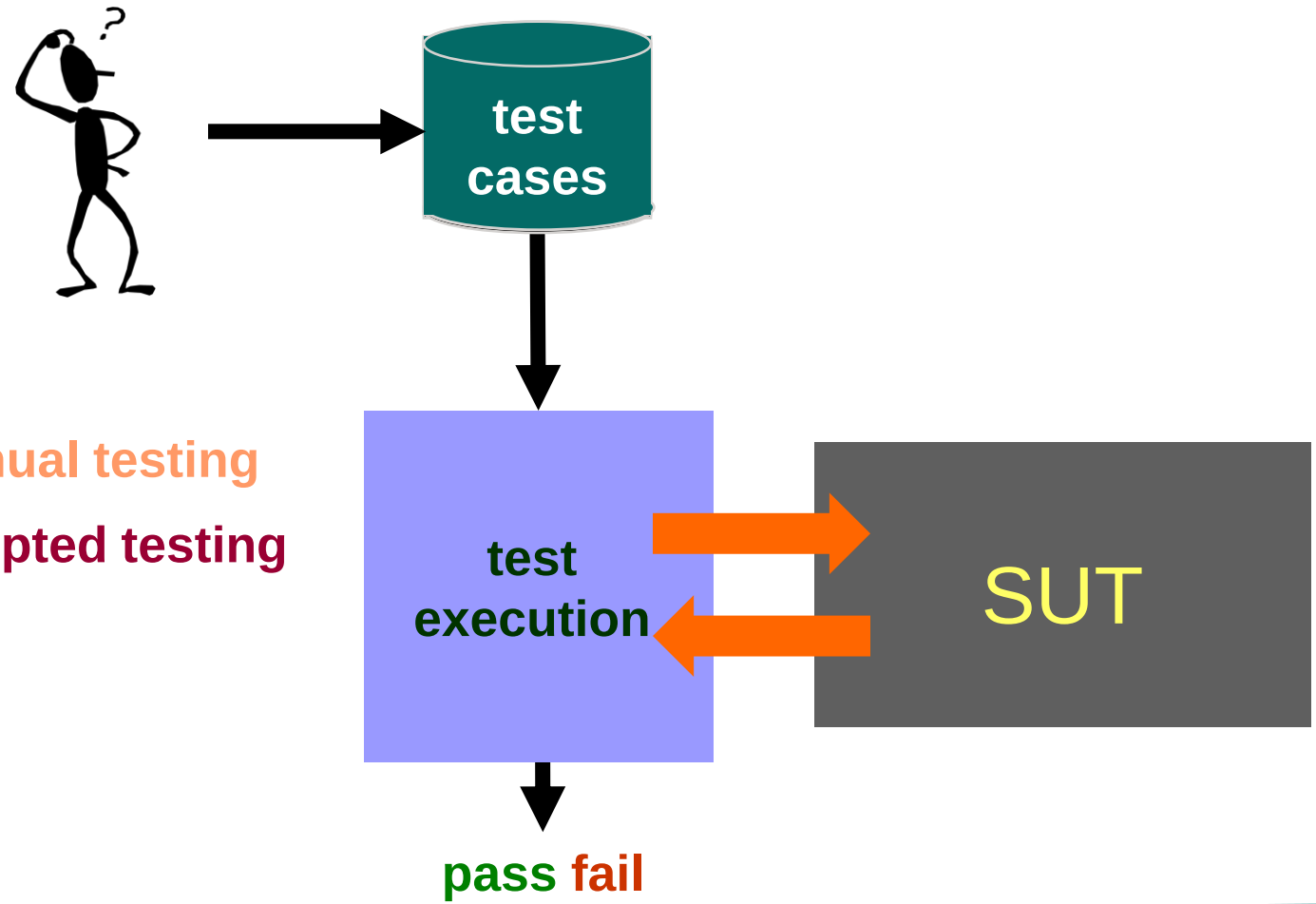


Developments in Testing 1

1. **Manual testing**

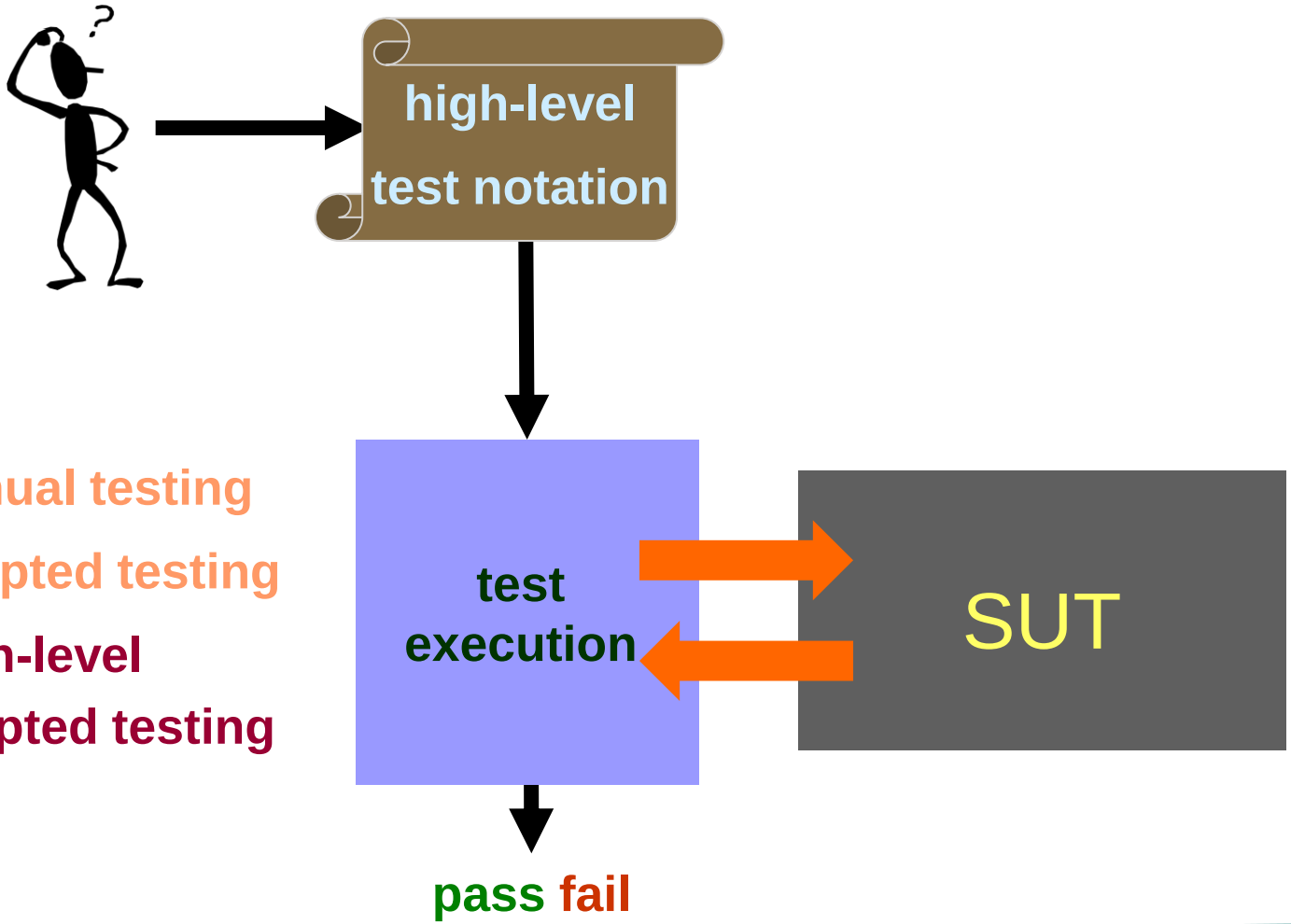


Developments in Testing 2



1. Manual testing
2. Scripted testing

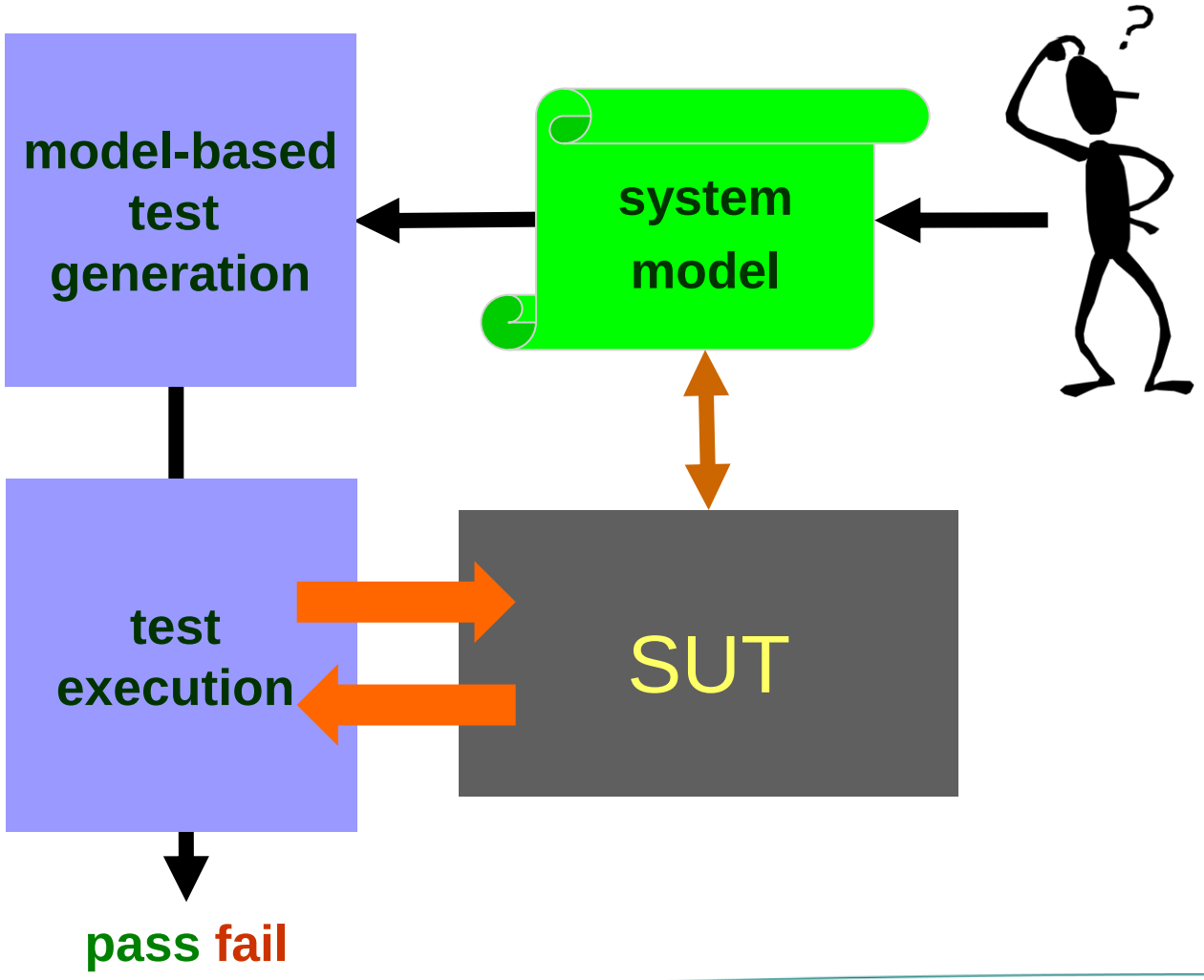
Developments in Testing 3



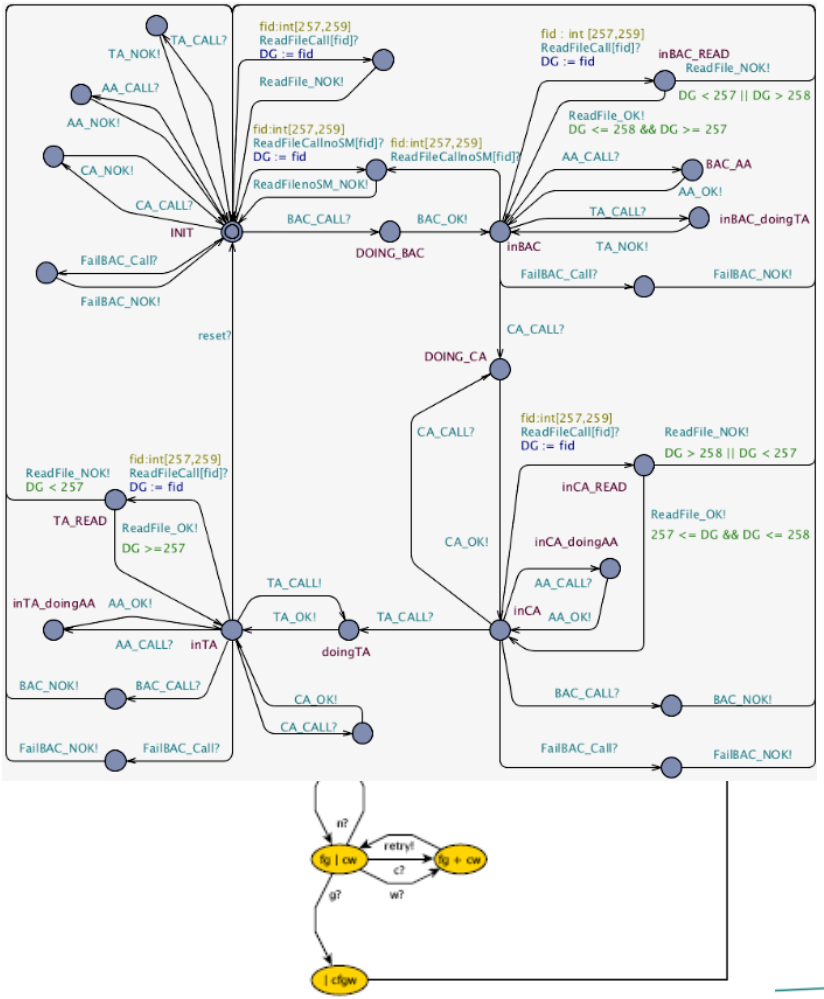
1. Manual testing
2. Scripted testing
3. **High-level scripted testing**

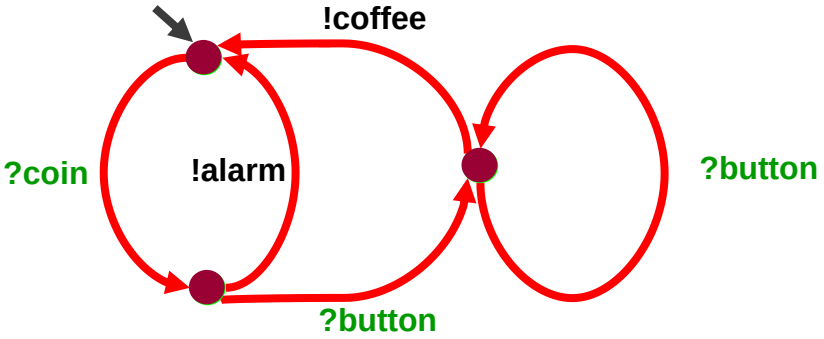
Developments in Testing 4

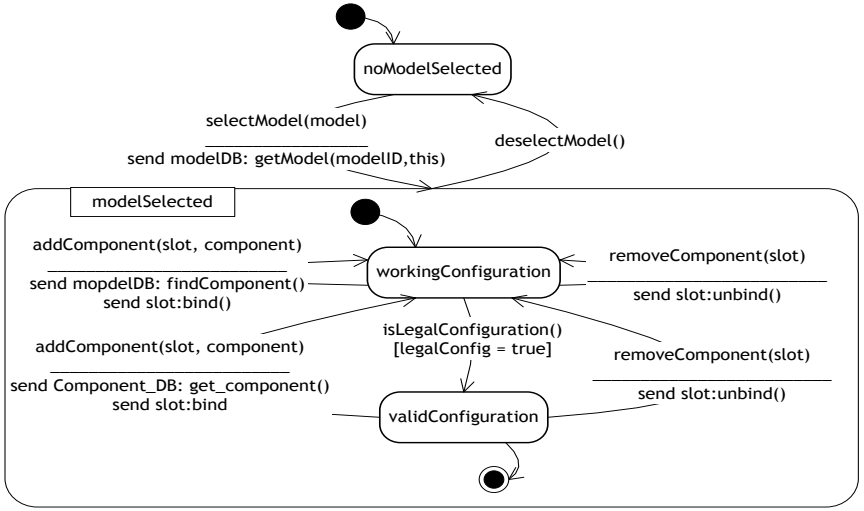
1. Manual testing
2. Scripted testing
3. High-level scripted testing
4. **Model-based testing**

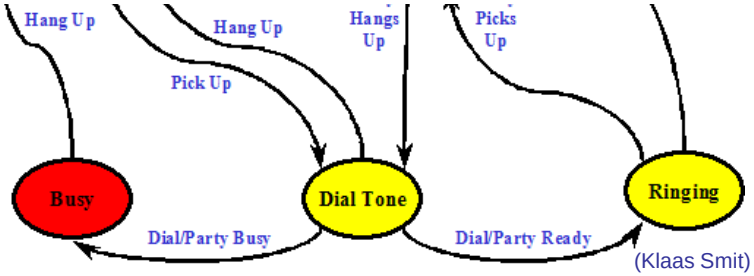


Models



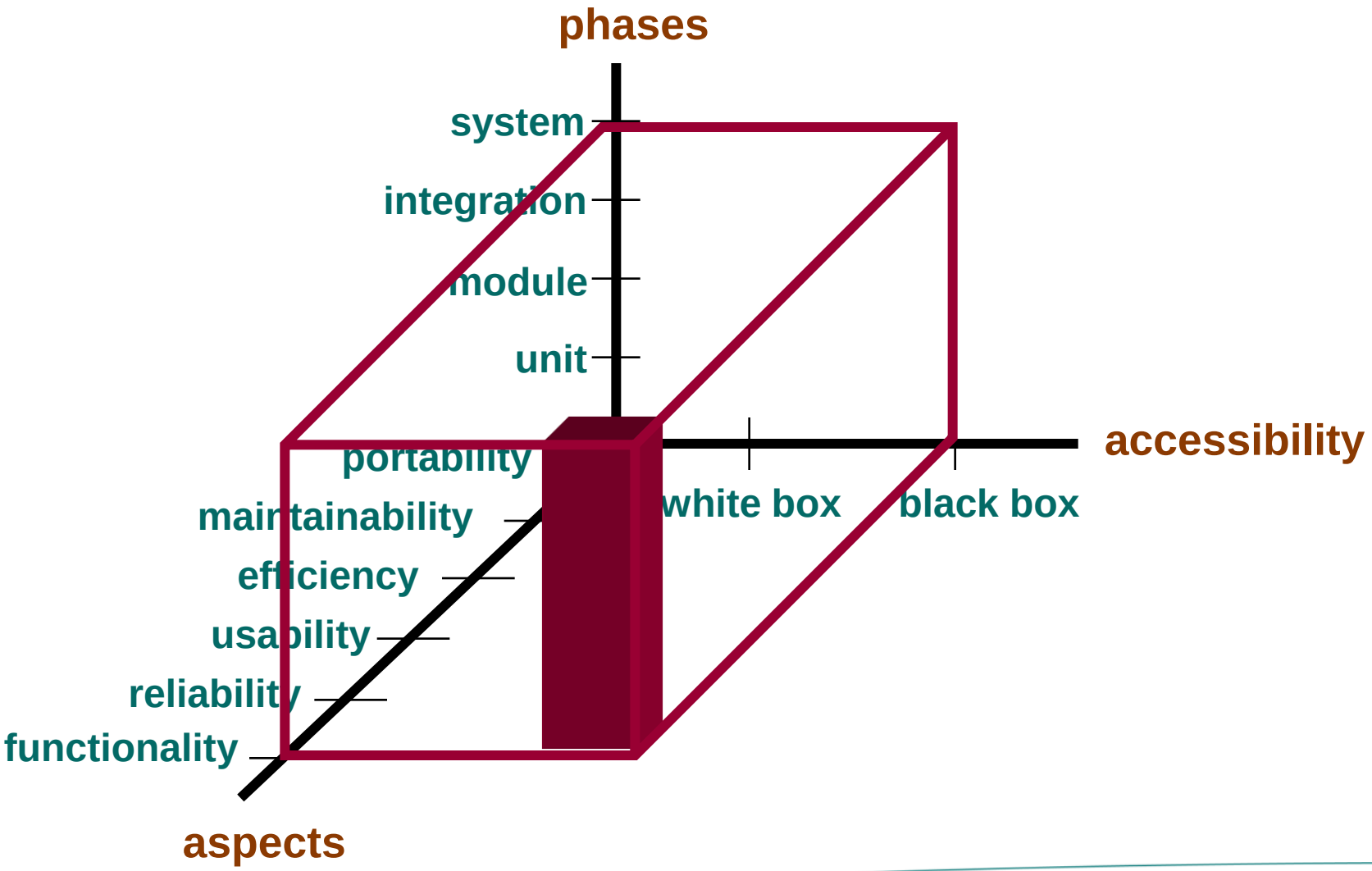






(Klaas Smit)

Sorts of Testing



Model-Based Testing: Why

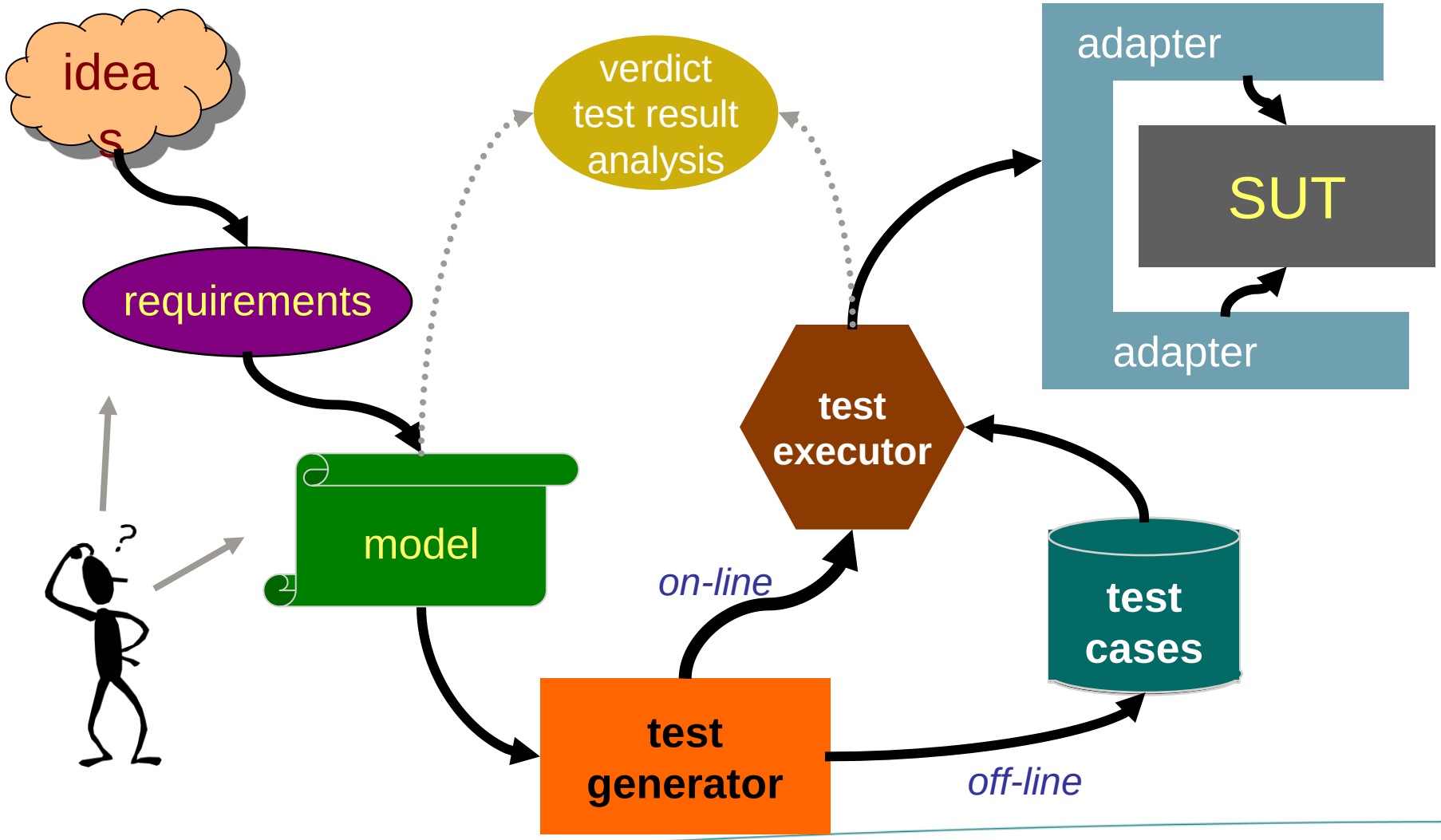
- Mastering increase in complexity, and quest for higher quality
  - testing cannot keep pace with development

Software bugs / errors cost US economy yearly:
$ 59.500.000.000 (www.nist.gov)
$ 22 billion could be eliminated…

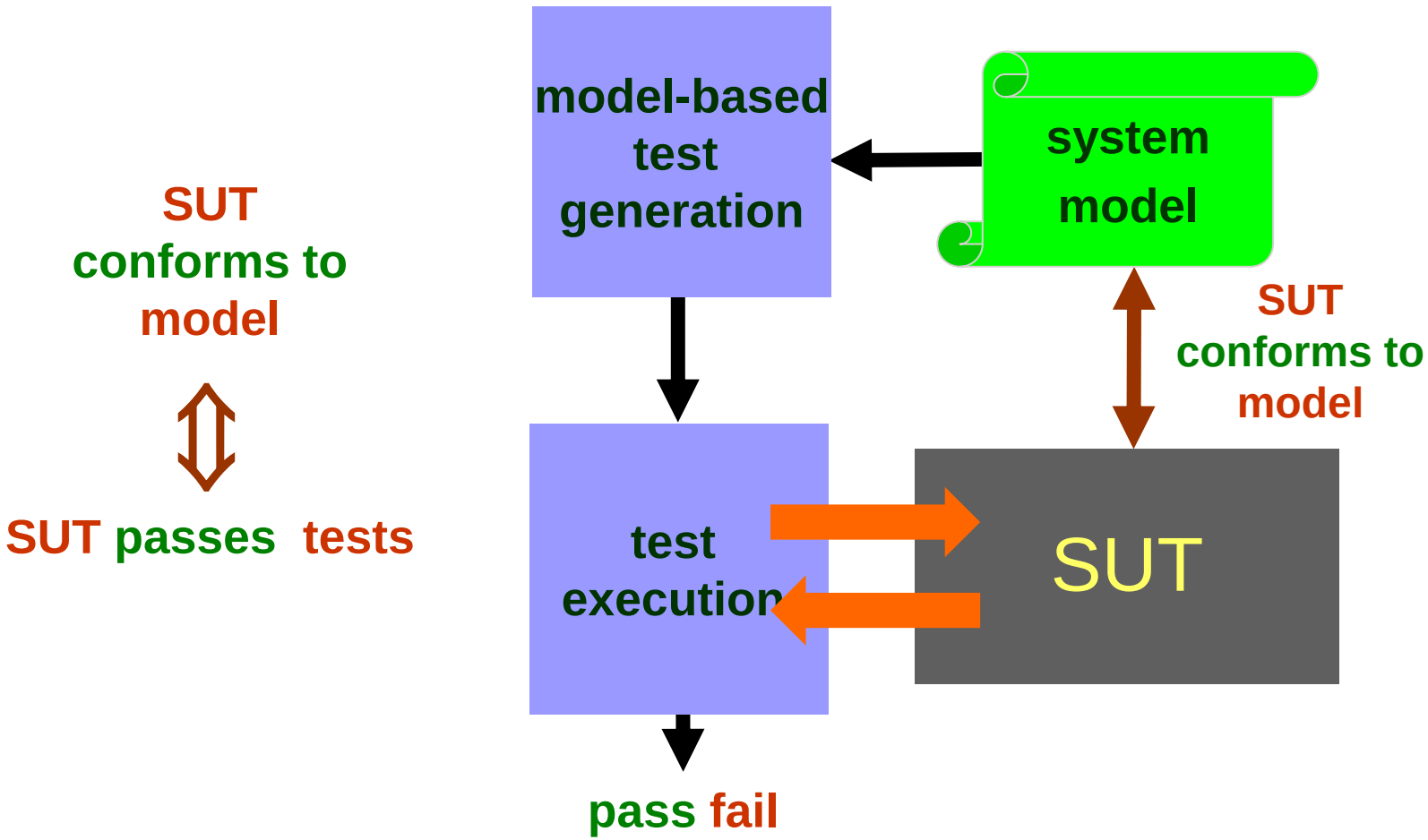
- Dealing with models and abstraction
  - model-based development: UML, MDA, Simulink/Matlab

- Promises better, faster, cheaper testing
  - algorithmic generation of tests and test oracles: tools
  - maintenance of tests through model modification

Model-Based Testing: The Process



Model-Based Testing: Some Theory

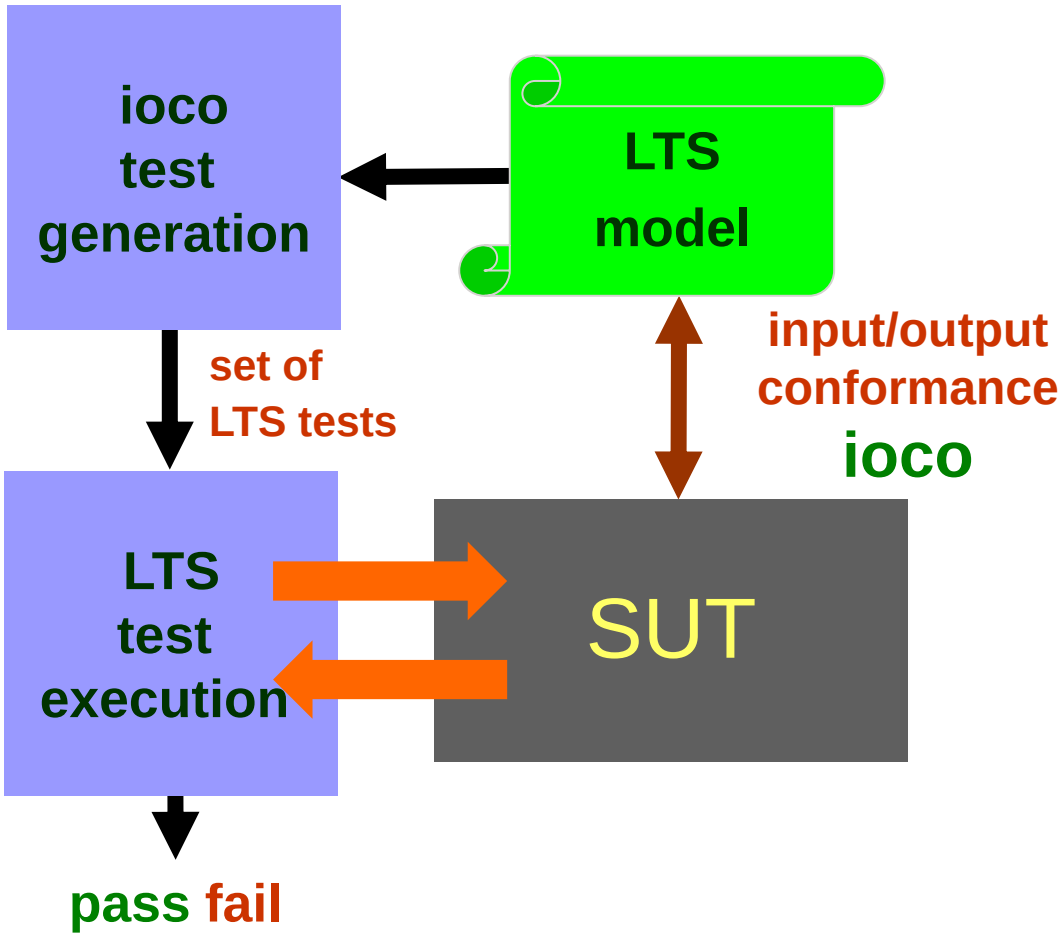


Model-Based Testing: Some Theory

SUT ioco model

⇕

SUT passes tests



Model-Based Testing: Some Tools

- AETG
- Agatha
- Agedis
- Autolink
- Conformiq Qtronic
- Cooper
- Uppaal-Cover
- G∀st
- Gotcha
- JTorX
- MaTeLo
- ParTeG
- Phact/The Kit
- QuickCheck
- Reactis
- RT-Tester
- SaMsTaG
- Smartesting Test Designer
- Spec Explorer
- Statemate
- STG
- TestGen (Stirling)
- TestGen (INT)
- TestComposer
- TGV
- TorX
- TorXakis
- T-Vec
- Uppaal-Tron
- Tveda
- . . . . . .

Model-Based Testing: Some Challenges

1. How to get a model
2. Adapter development
3. Test selection, a-priori coverage
4. Quality of tested systems, posterior coverage
5. Relation to other model-based activities, diagnosis
6. Non-functional testing: performance, security, ...
7. Integration in the testing process
8. Education for MBT
9. Scalability
10. ROI: Return on Investment

And now:

1. *Neda Noroozi*

   Model-based testing of electronic funds transfer systems

2. *Axel Belinfante*

   Model-based testing of a wireless sensor network node

3. *Marten Sijtema*

   Experiences with formal engineering: Model-based specification, implementation, and testing of a software bus