Radboud Universiteit

ESI

Model-Based Testing

Jan Tretmans

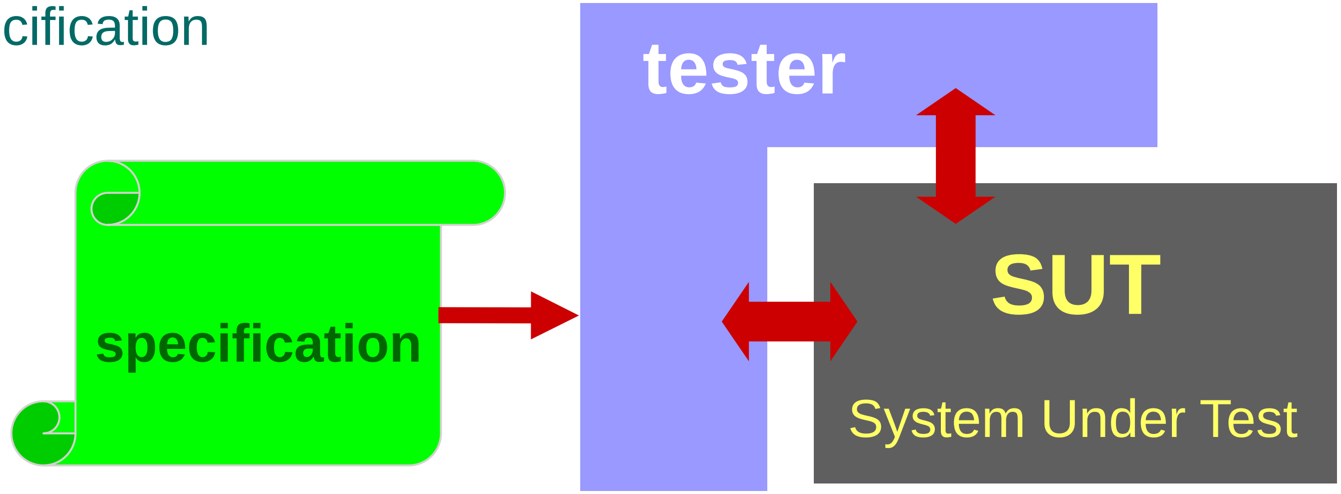
ESI – Embedded Systems Innovation by TNO
Radboud University Nijmegen
Högskolan i Halmstad
jan.tretmans@tno.nl

TNO innovation for life

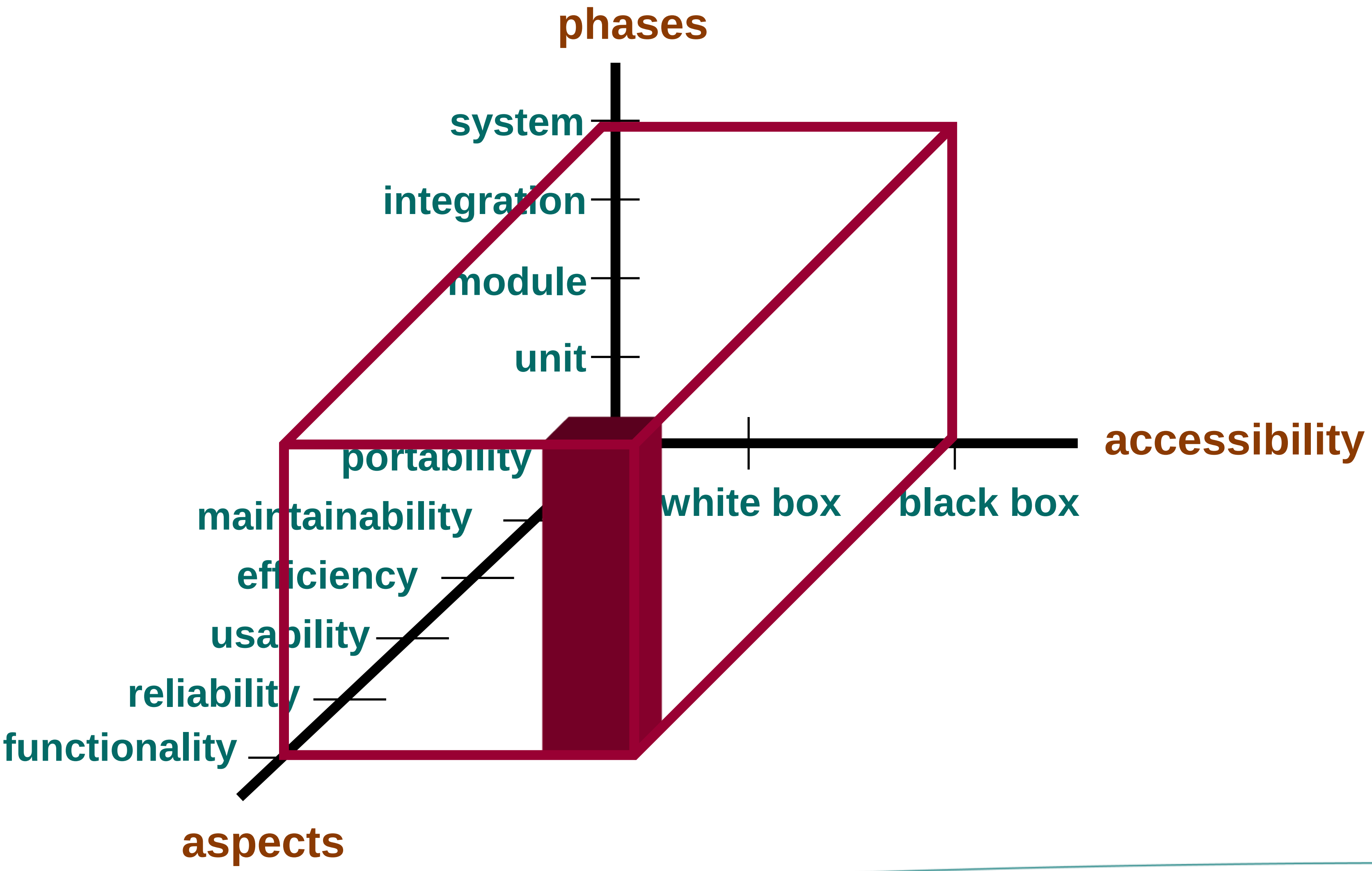
Software Testing

Checking or measuring
some quality characteristics
of an executing software object
by performing experiments
in a controlled way
w.r.t. a specification

*specification-based,
active, black-box
testing of
functionality*



MBT : Black-Box Testing of Functionality



Model-Based Testing

Basics

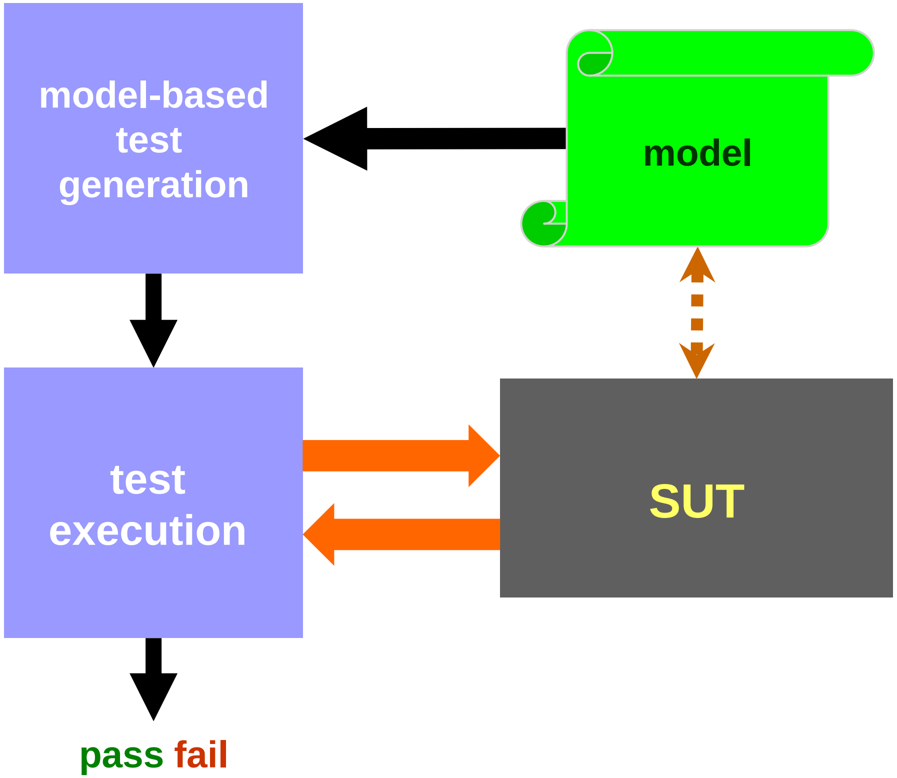
Model-Based Testing

MBT

next step in

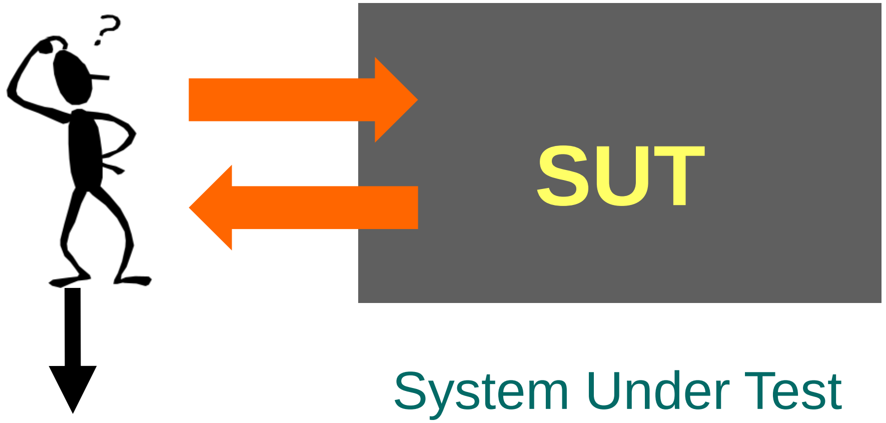
test automation:

- + test generation
- + result analysis



1 : Manual Testing

1. **Manual testing**

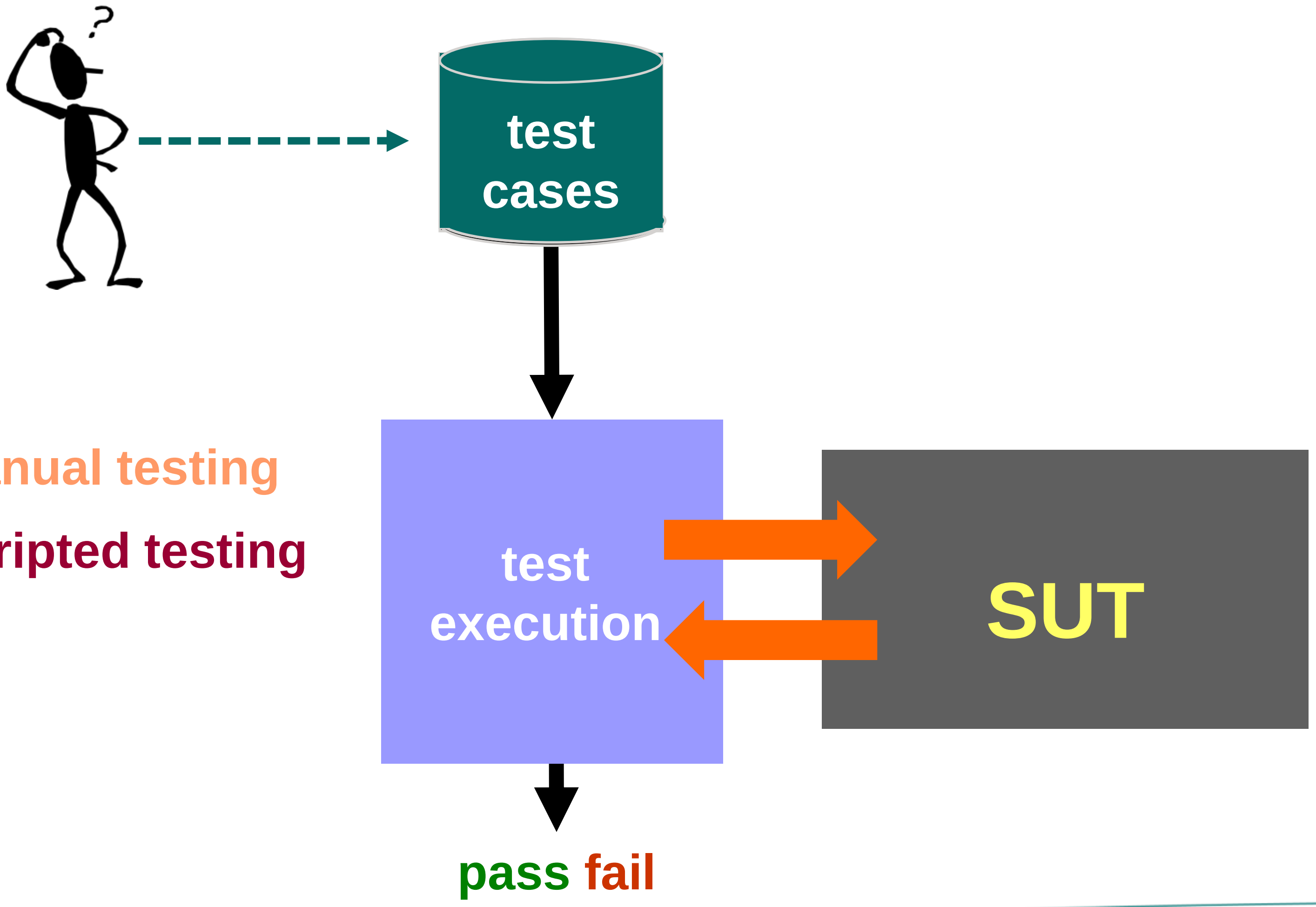


SUT

pass fail

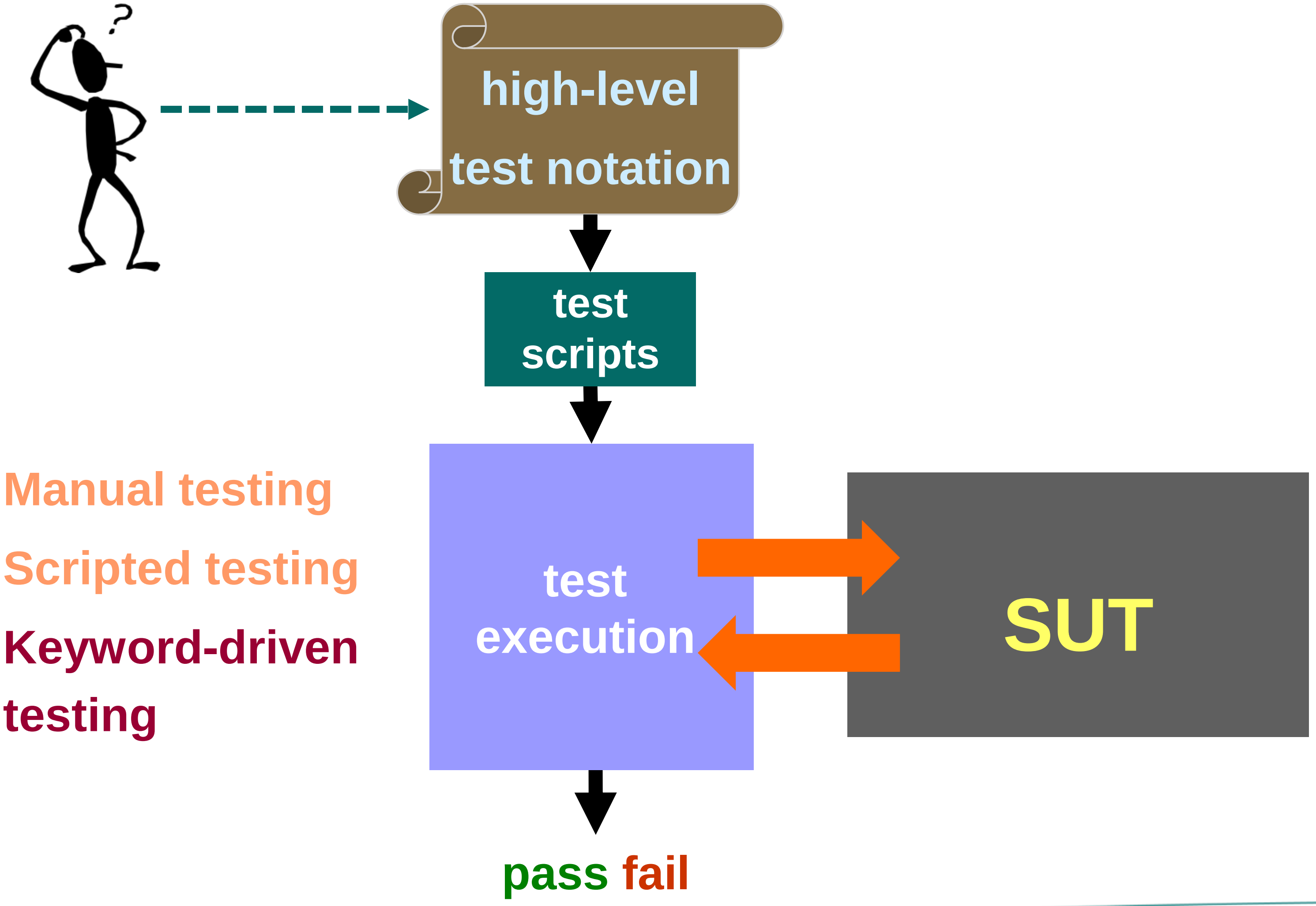
System Under Test

2 : Scripted Testing



1. Manual testing
2. Scripted testing

3 : Keyword-Driven Testing



high-level test notation

test scripts

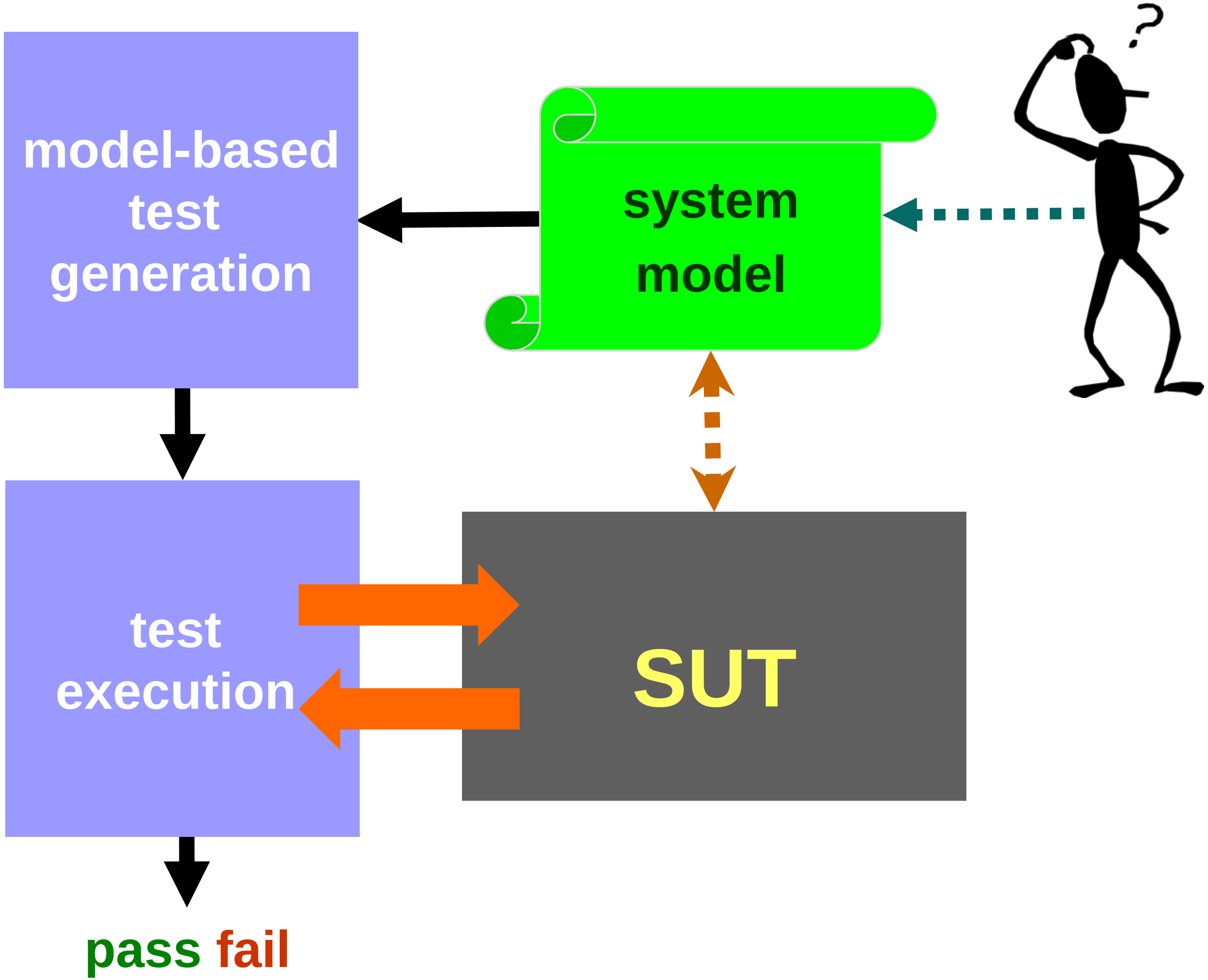
test execution

SUT

pass fail

1. Manual testing
2. Scripted testing
3. Keyword-driven testing

4 : Model-Based Testing



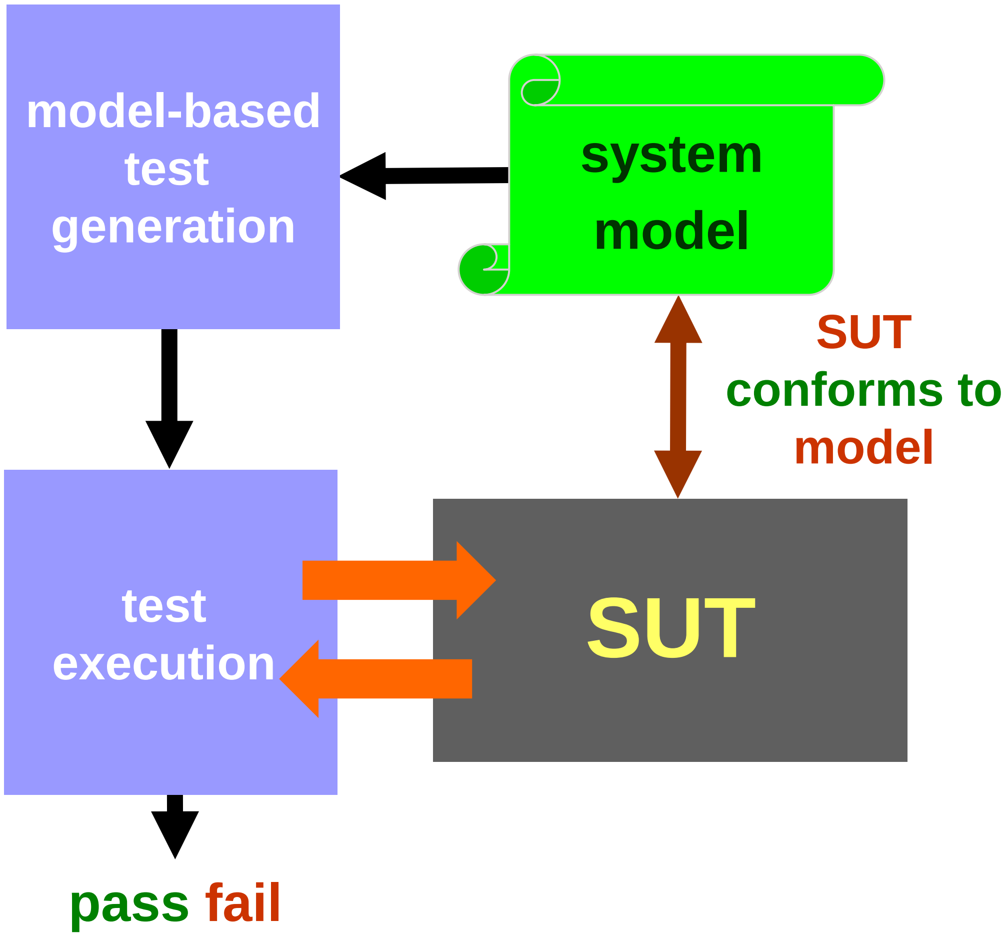
1. Manual testing
2. Scripted testing
3. Keyword-driven testing
4. Model-based testing

Model Based Testing

SUT
conforms to
model

sound ⇓ ⇑ *exhaustive*

SUT passes tests



model-based test generation

system model

SUT conforms to model

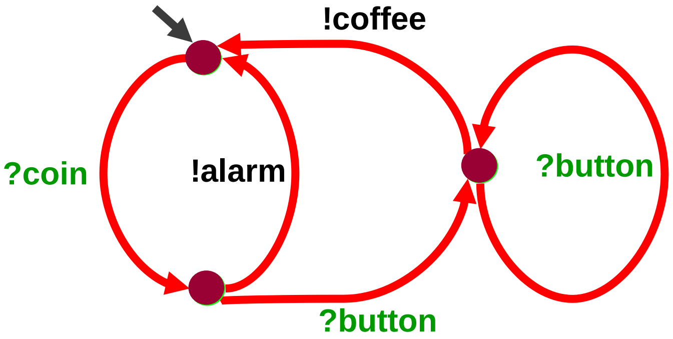
test execution

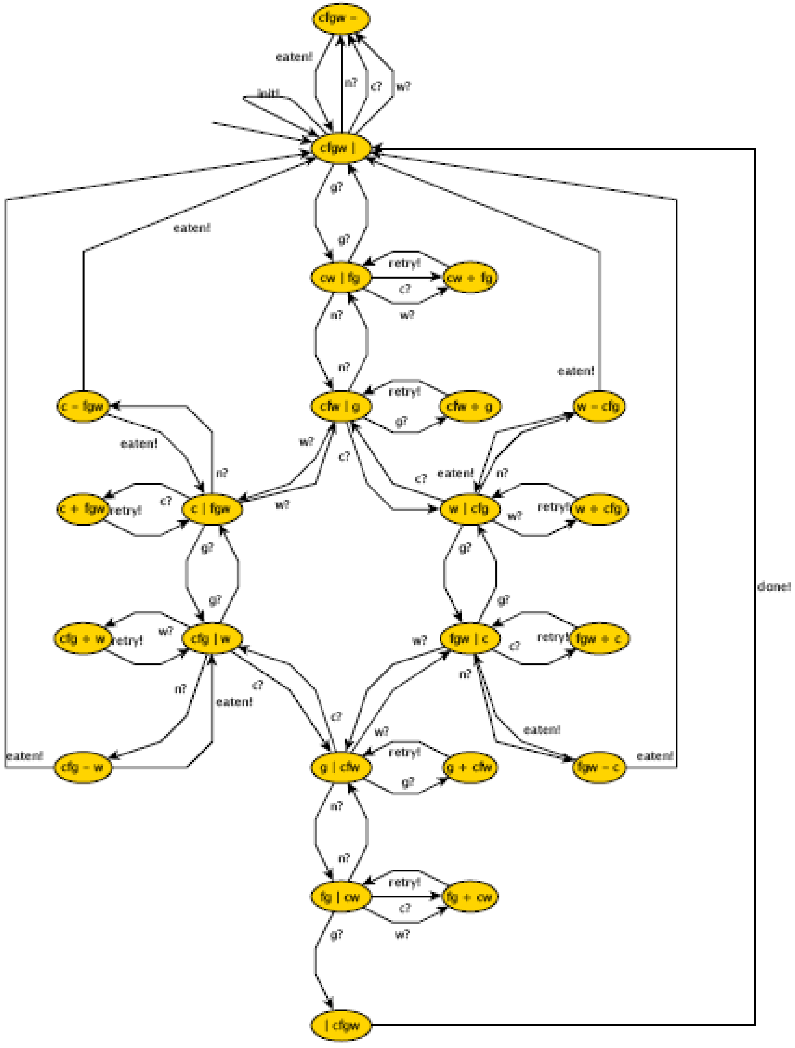
SUT

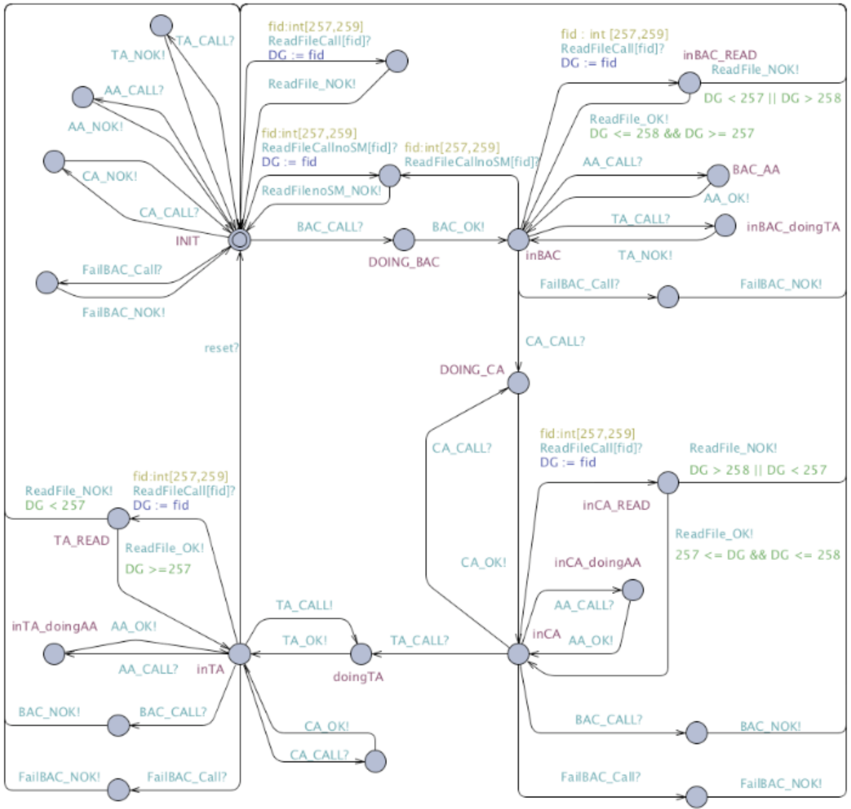
pass fail

Models

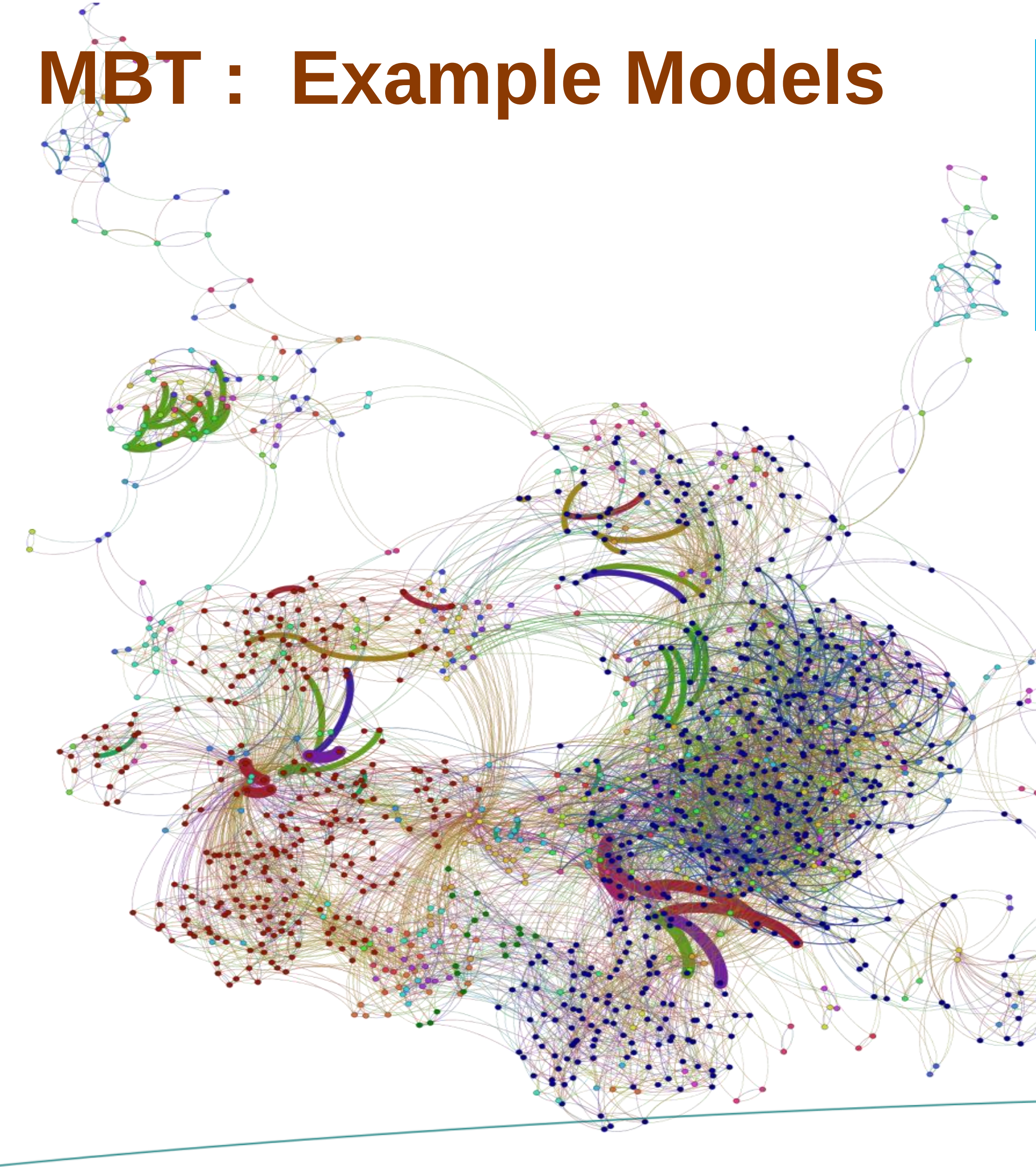
MBT : Example Models





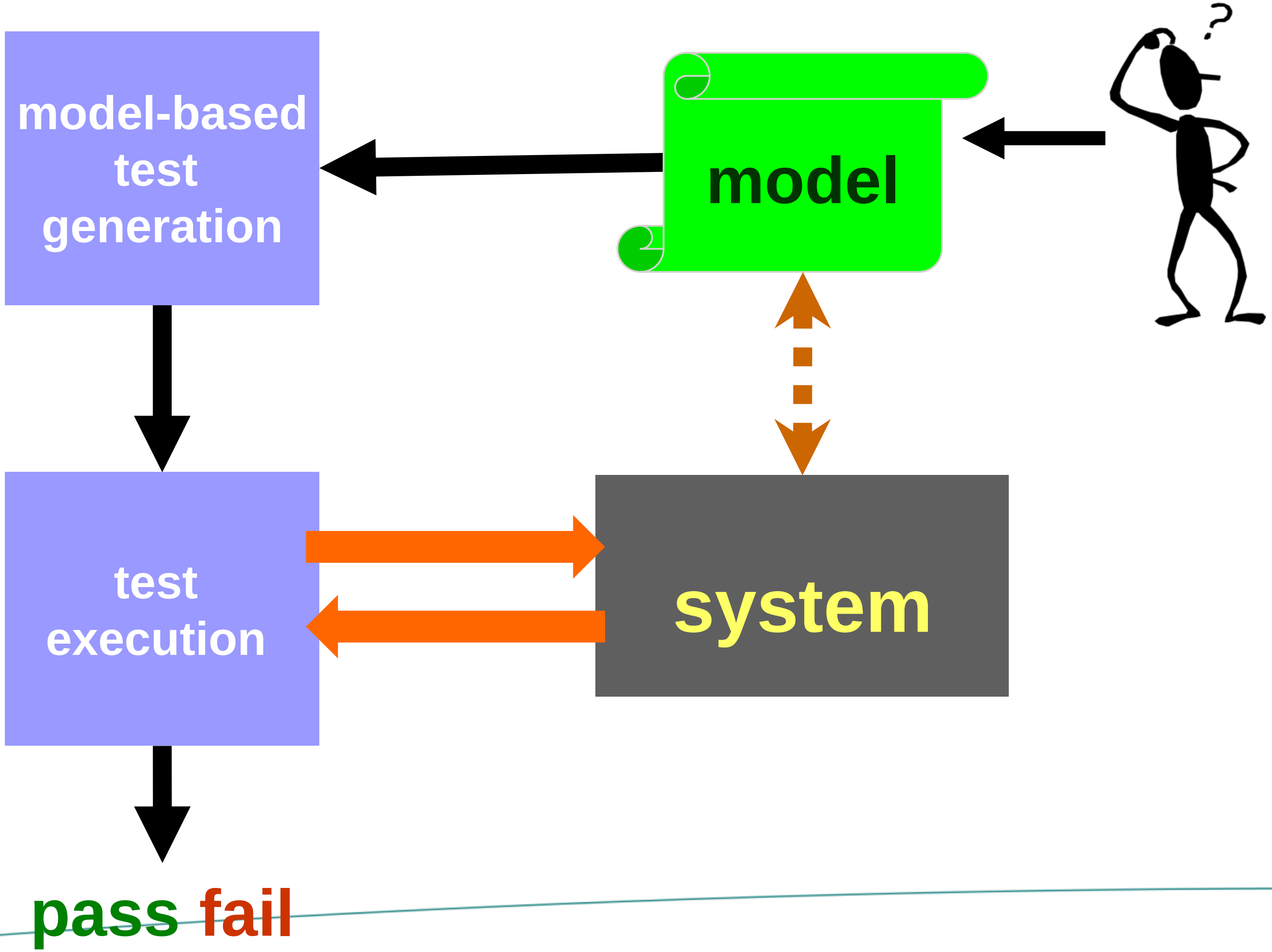


MBT : Example Models

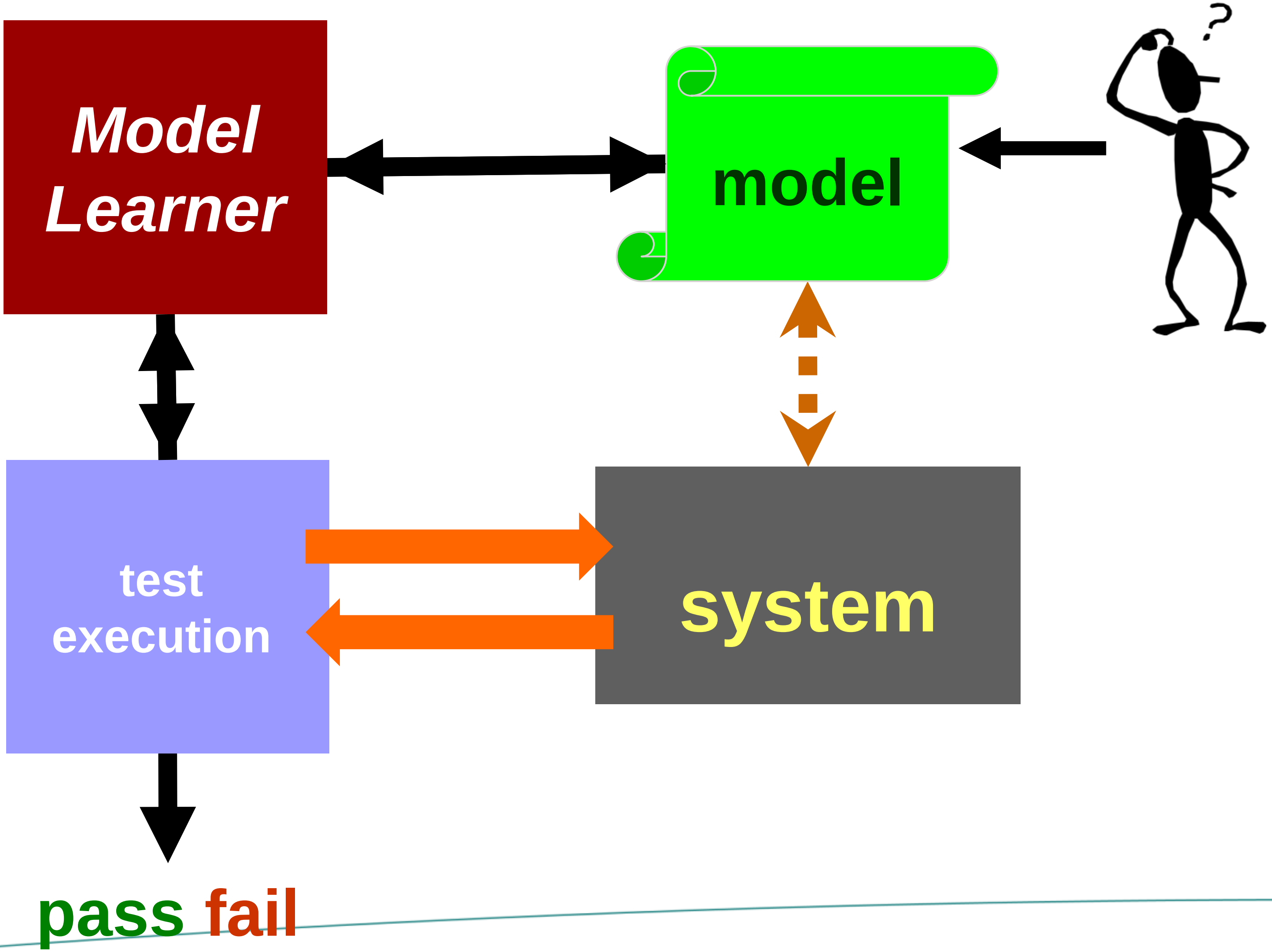


Model Learning

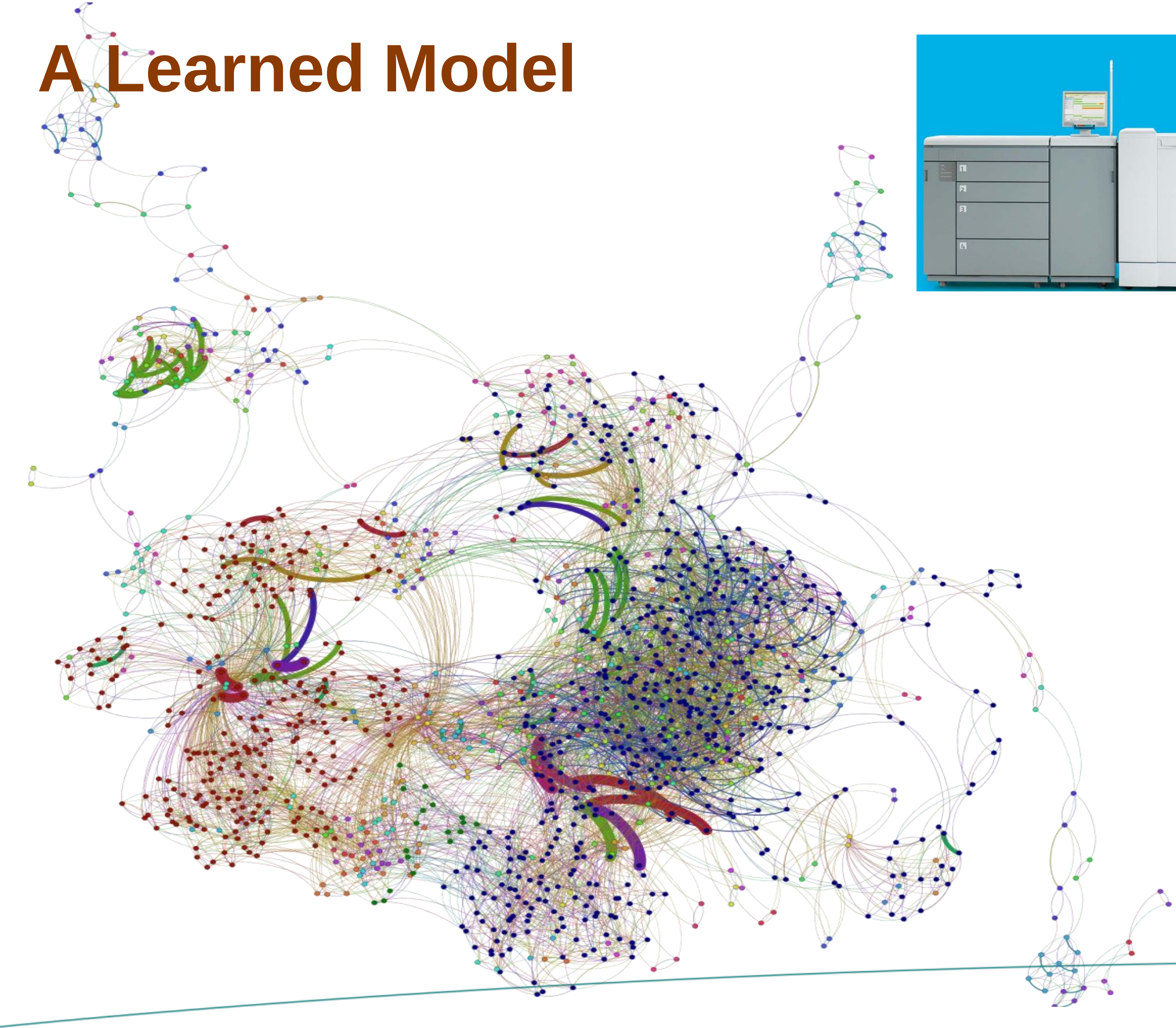
MBT : How to Get these D… Models



Research : Model Learning



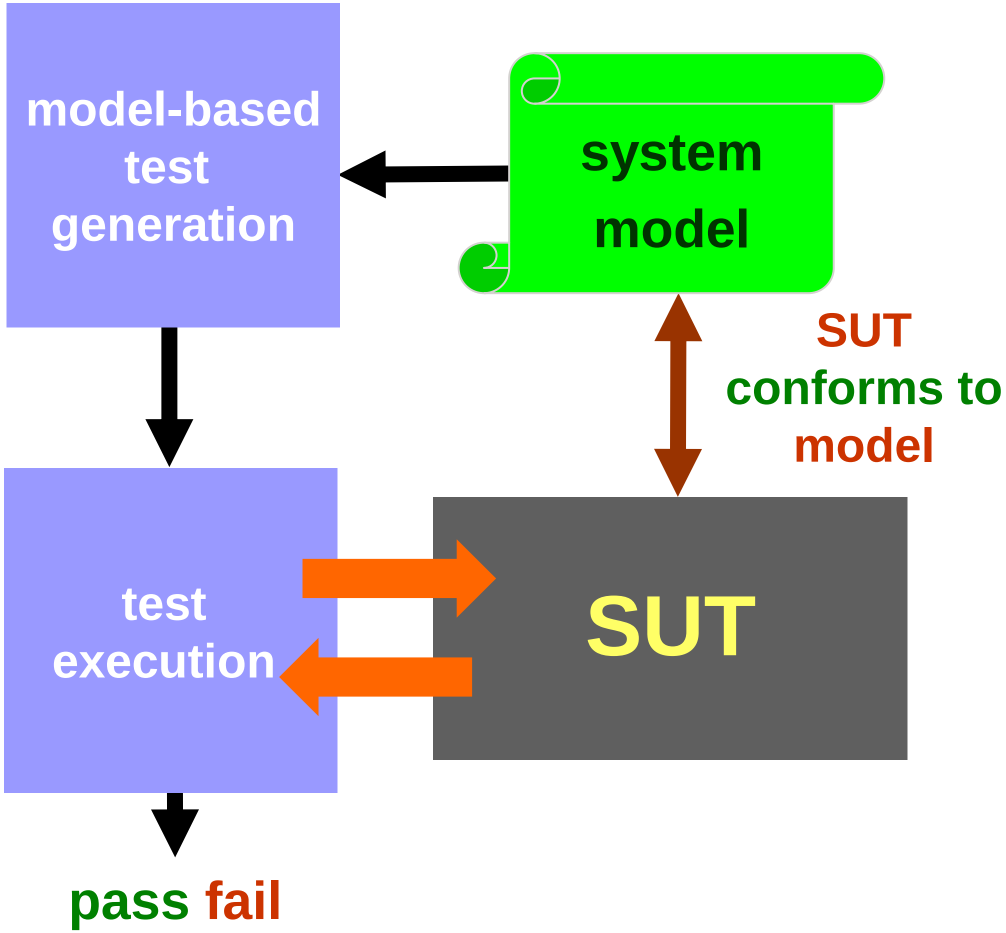
A Learned Model



Model-Based Testing

Theory

Model Based Testing



SUT conforms to model

sound ⇓ ⇑ *exhaustive*

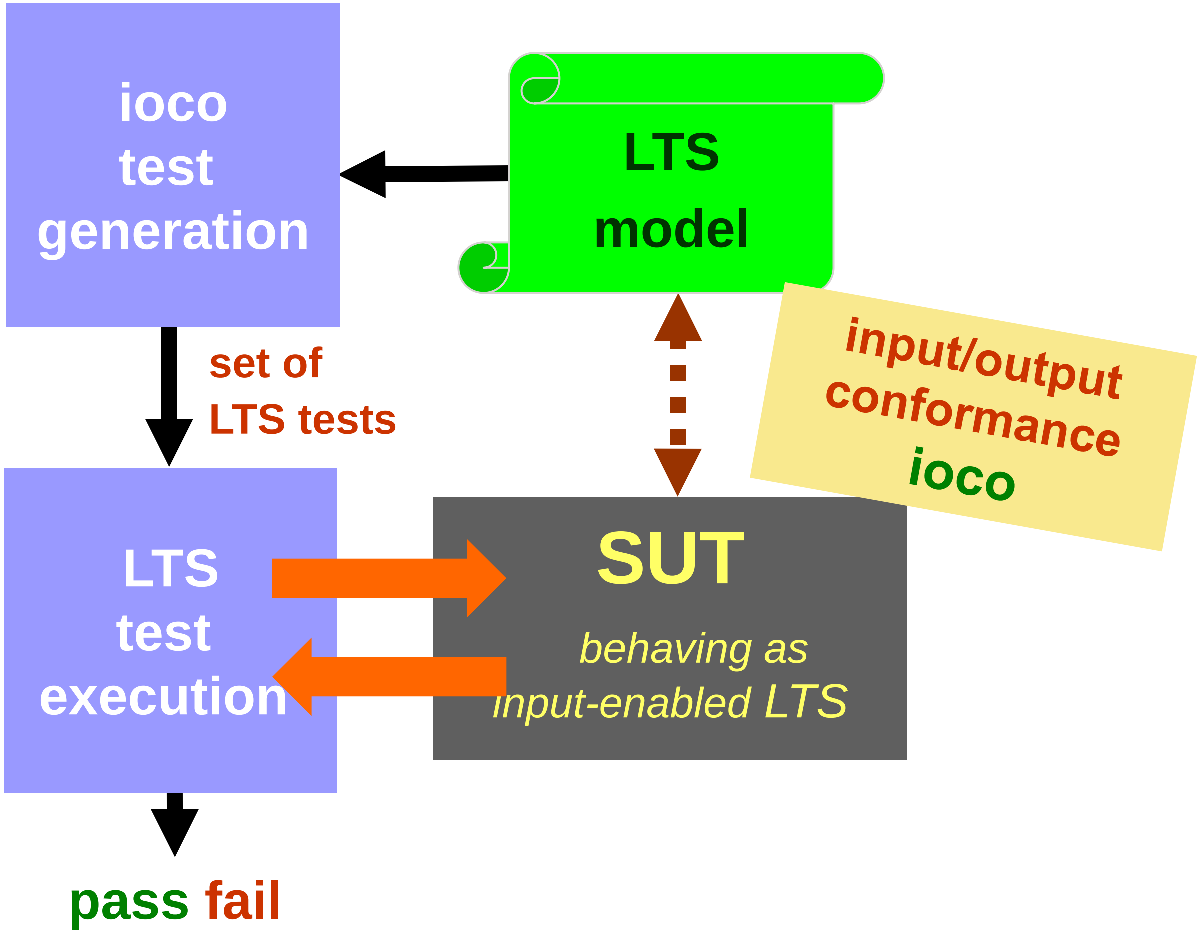
SUT passes tests

MBT : LTS & ioco

SUT ioco model

sound ⇓ ⇑ *exhaustive*

SUT passes tests



ioco test generation

LTS model

set of LTS tests

input/output conformance ioco

LTS test execution

SUT

behaving as input-enabled LTS

pass fail

Model-Based Testing Theory
with Labelled Transition Systems
and **ioco**

Conformance : *ioco*

$$i \ \ \mathbf{ioco} \ \ s \quad =_{def} \ \forall \sigma \in \mathit{Straces}\,(s) : \ \mathit{out}\,(i \text{ after } \sigma) \ \subseteq \ \mathit{out}\,(s \text{ after } \sigma)$$

s is a Labelled Transition System (LTS)

i is (assumed to be) an input-enabled LTS

$$p \xrightarrow{\delta} p \quad = \quad \forall \ \ !x \ \in \ L_U \cup \{\tau\} \ . \ \ p \not\xrightarrow{!x}$$

$$\mathit{Straces}\,(\,s\,) \quad = \quad \{ \ \sigma \ \in \ (\,L \cup \{\delta\}\,)^* \ \mid \ s \overset{\sigma}{\Longrightarrow} \ \}$$

$$p \text{ after } \sigma \quad = \quad \{ \ p' \mid \ p \overset{\sigma}{\Longrightarrow} \ p' \ \}$$

$$\mathit{out}\,(\,P\,) \ = \{ \ !x \in L_U \mid p \xrightarrow{!x} , \ p \in P \,\} \ \cup \ \{\, \delta \mid p \xrightarrow{\delta} p, \ \ p \in P \,\}$$

Conformance : *ioco*

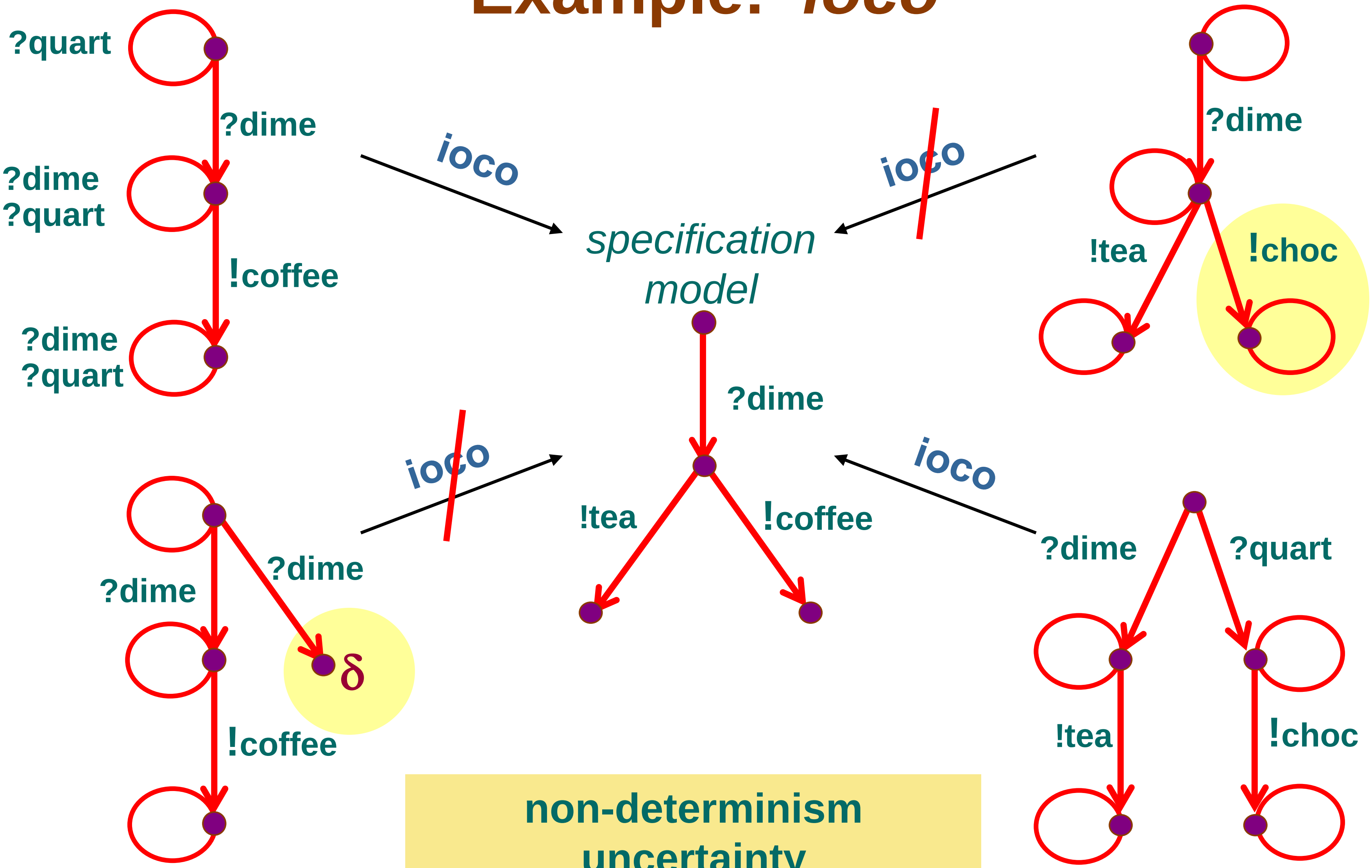
$$\mathbf{i \ \ ioco \ \ s \quad =_{def} \ \forall \sigma \in \textit{Straces}\ (s) : \ \textit{out}\ (i\ after\ \sigma) \ \subseteq \ \textit{out}\ (s\ after\ \sigma)}$$

Intuition:

i ioco-conforms to s, iff

- if i produces output x after trace σ,
  then s can produce x after σ
- if i cannot produce any output after trace σ,
  then s cannot produce any output after σ (*quiescence* δ)

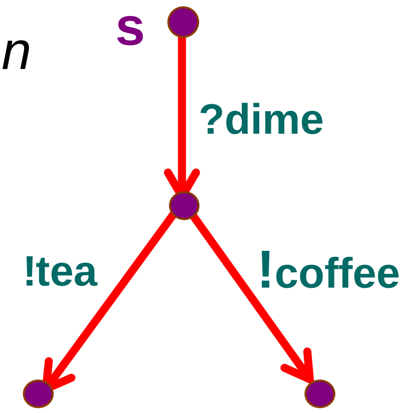
Example: *ioco*

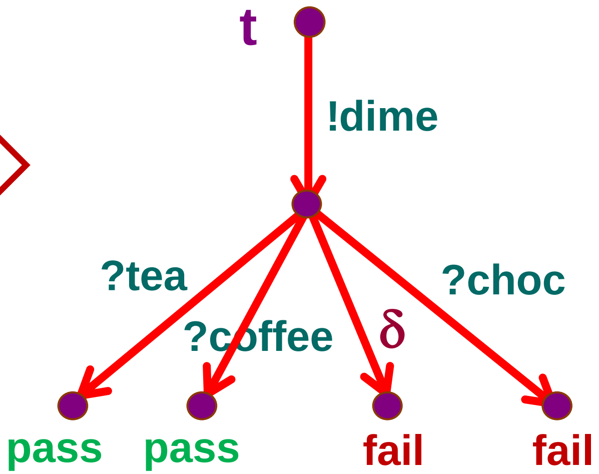


non-determinism
uncertainty
under-specification

Example: *ioco* Test Generation

specification model

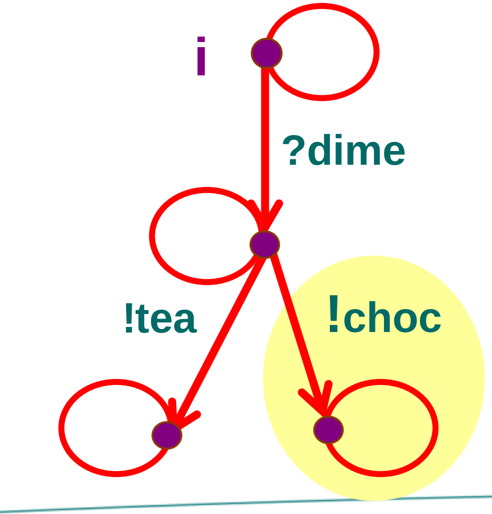




generated test case

i ~~ioco~~ s

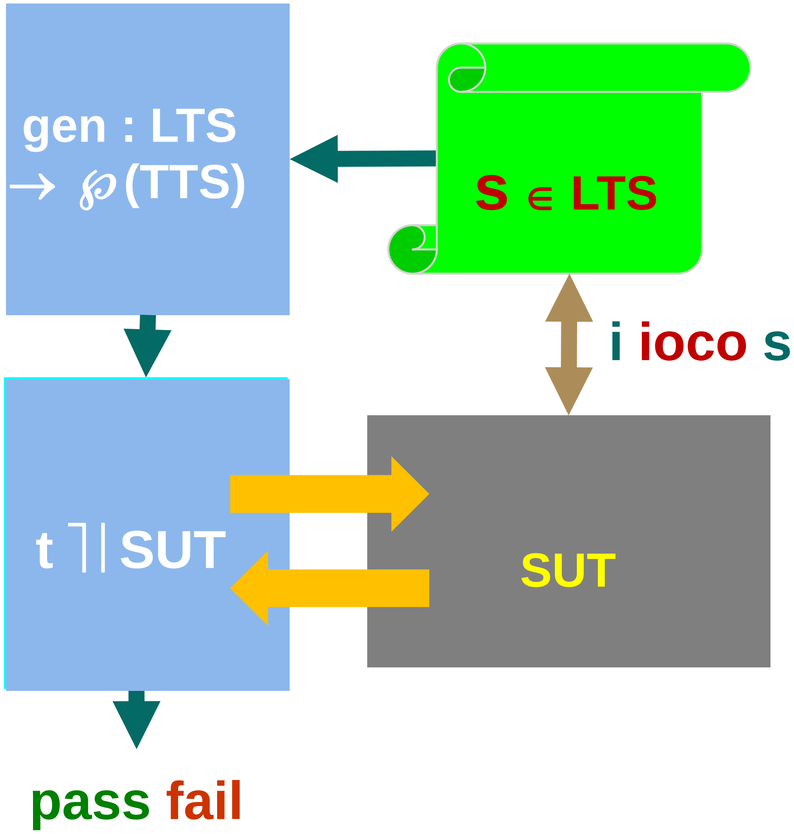
i fails t



implementation

i ioco s $=_{def}$
∀ σ ∈ *Straces* (s) :
out (i after σ) ⊆ *out* (s after σ)

MBT with *ioco* is Sound and Exhaustive



Test assumption :

$\forall \mathbf{SUT} \in \mathrm{IMP}\ .\ \exists \mathbf{m}_{\mathbf{SUT}} \in \mathrm{IOTS}\ .$

$\quad \forall t \in \mathrm{TESTS}\ .$

$\qquad \mathbf{SUT}\ \mathbf{passes}\ t \Leftrightarrow \mathbf{m}_{\mathbf{SUT}}\ \mathbf{passes}\ t$

Prove soundness and exhaustiveness:

$\forall \mathbf{m} \in \mathrm{IOTS}\ .$

$\qquad (\ \forall t \in \mathrm{gen}(\mathbf{s})\ .\ \mathbf{m}\ \mathbf{passes}\ t\ )$

$\Leftrightarrow \quad \mathbf{m}\ \mathbf{ioco}\ \mathbf{s}$

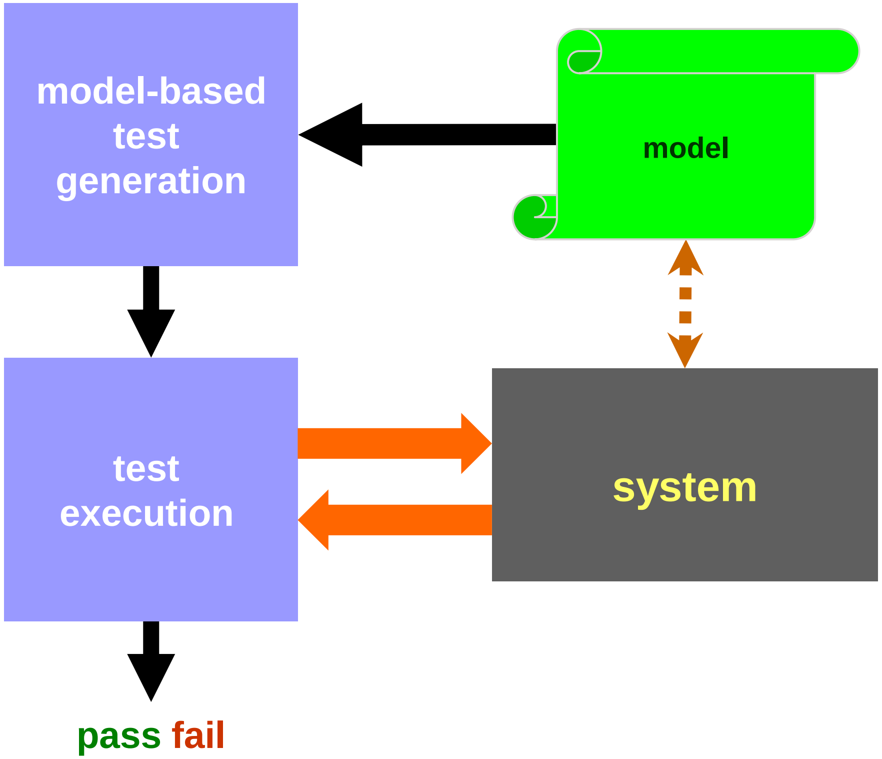
SUT comforms to s

exhaustive ⇑ ⇓ *sound*

SUT passes gen(s)

MBT

?



model-based test generation

model

test execution

system

pass fail