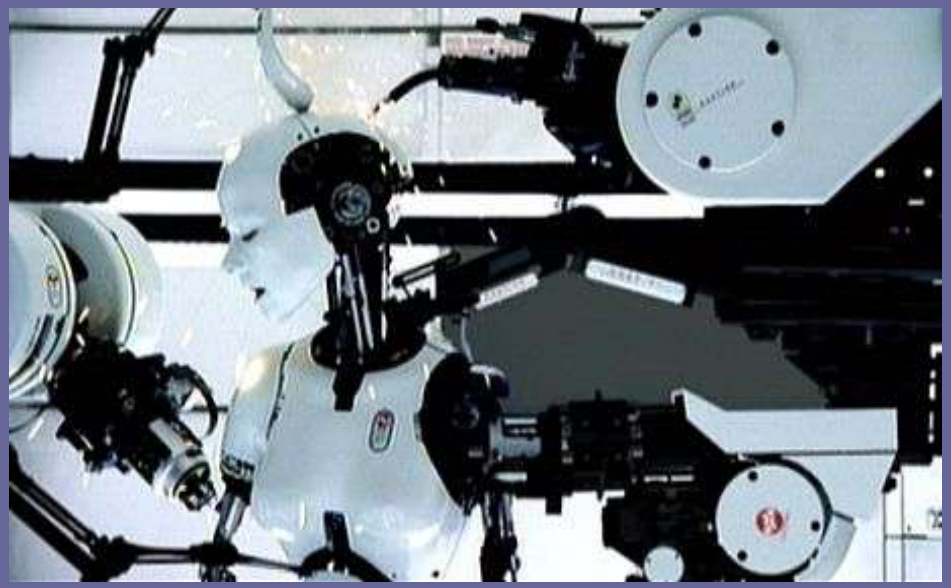
Department seminar
University of Bergen
May 13 - 2004



Model-Based Testing
An Overview

Lars Frantzen
lf@cs.kun.nl
www.cs.kun.nl/~lf/

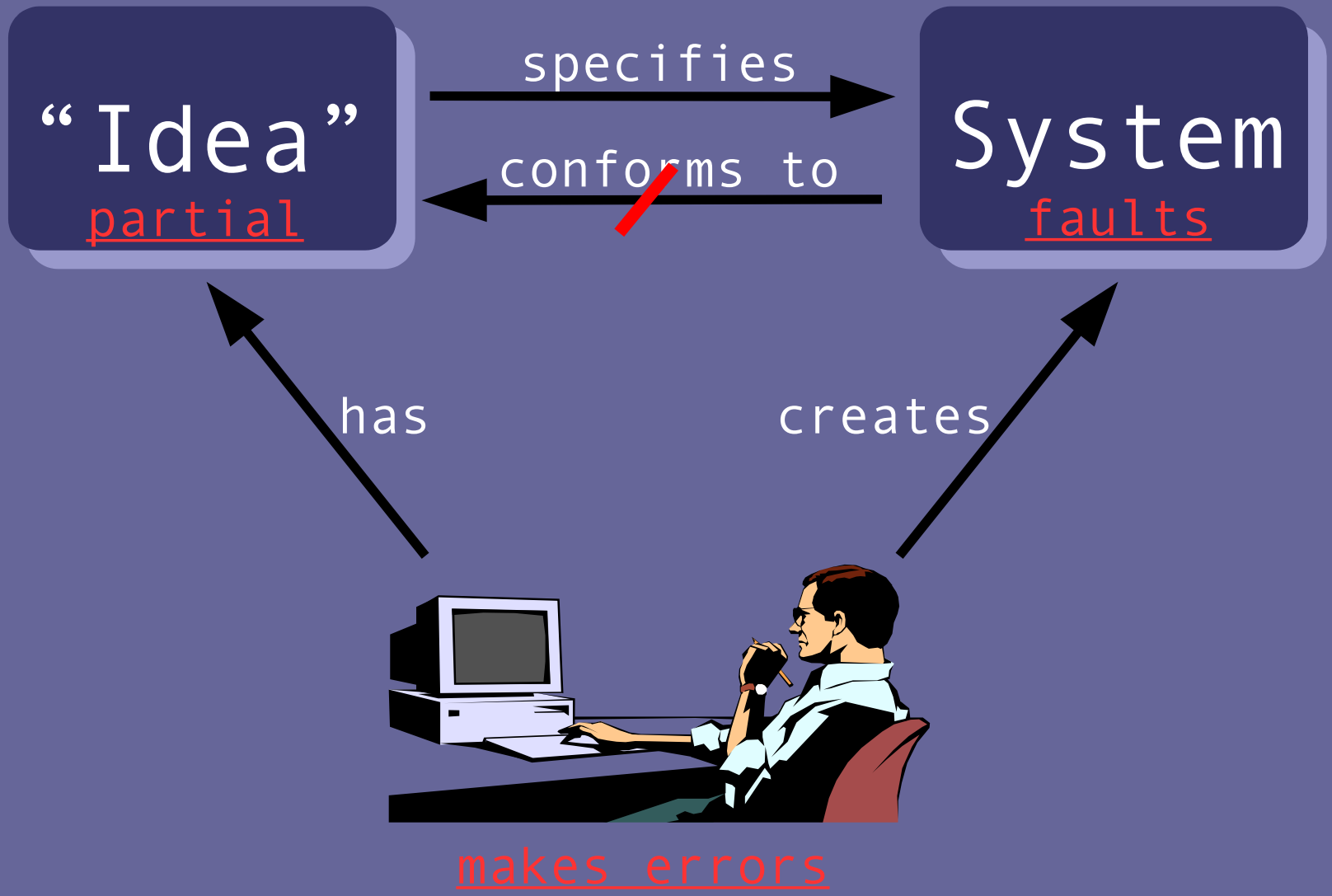
Radboud University Nijmegen

Ideology

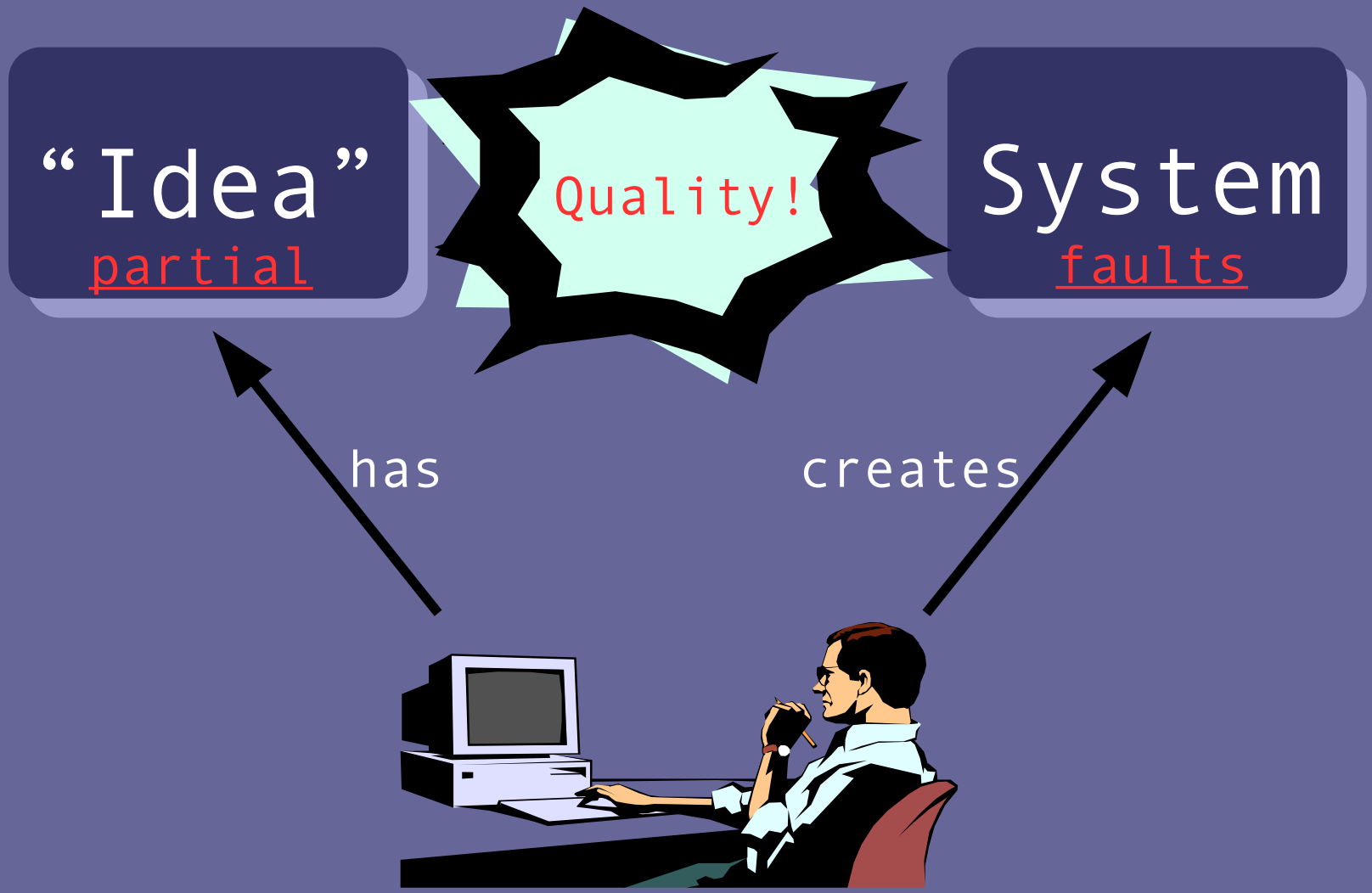
Plato

The world that appears to our senses is in some way defective and filled with error, but there is a more real and perfect realm, populated by entities (called “forms” or “ideas”) that are eternal, changeless, and in some sense paradigmatic for the structure and character of our world.

"Ideas", Systems, and the Human Factor



"Ideas", Systems, and the Human Factor



Quality Aspects

ISO 9126:

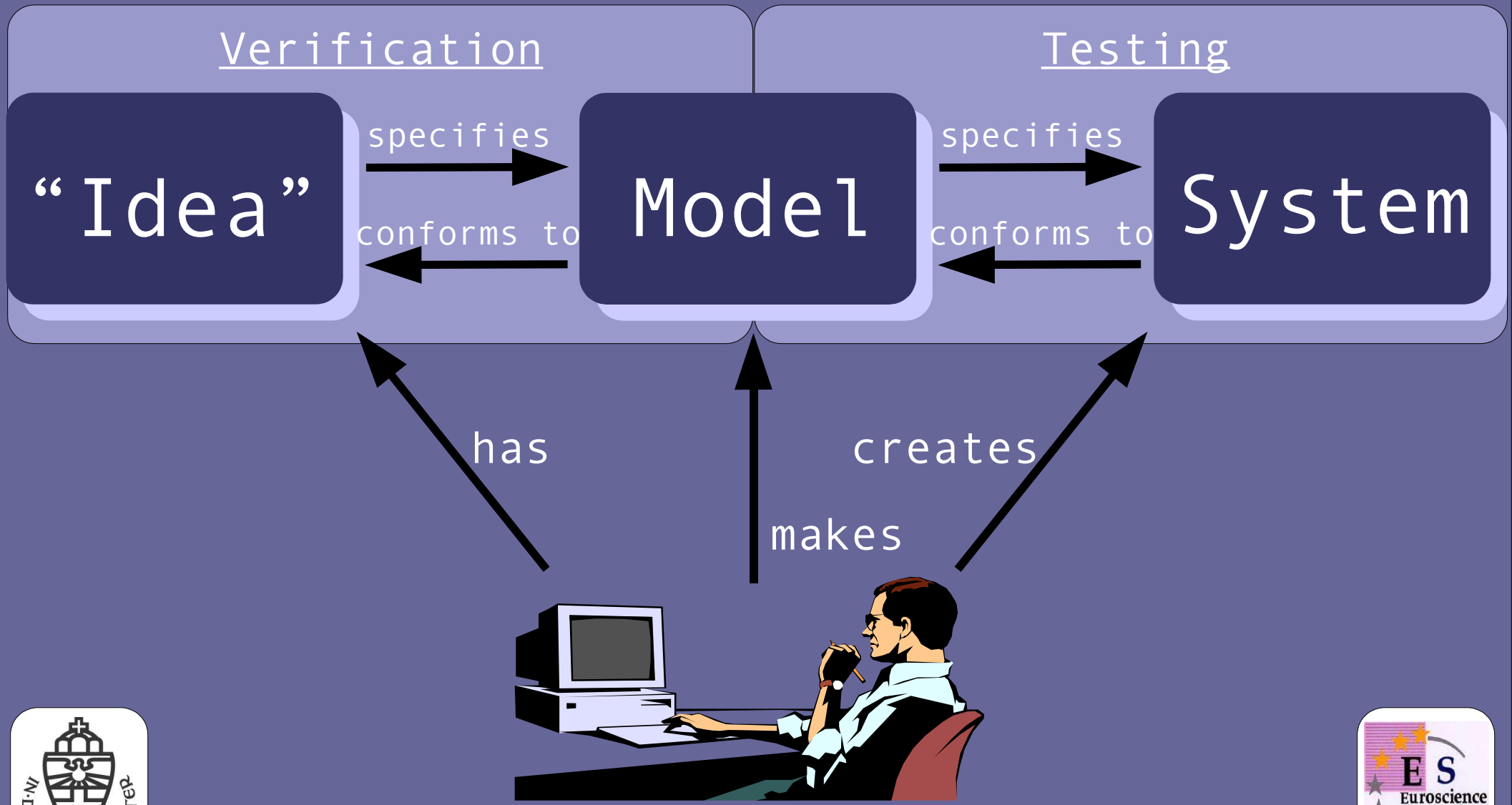
- Functionality
- Reliability
- Usability
- Efficiency
- Maintainability
- Portability / Applicability

Quality measures the degree of conformance with aspects of an "idea" of the system.

To check and improve quality, these aspects have to be formalised, i.e. modeled.

These aspects can not be modeled exhaustively due to complexity.
→ Abstraction

Model-Based Development



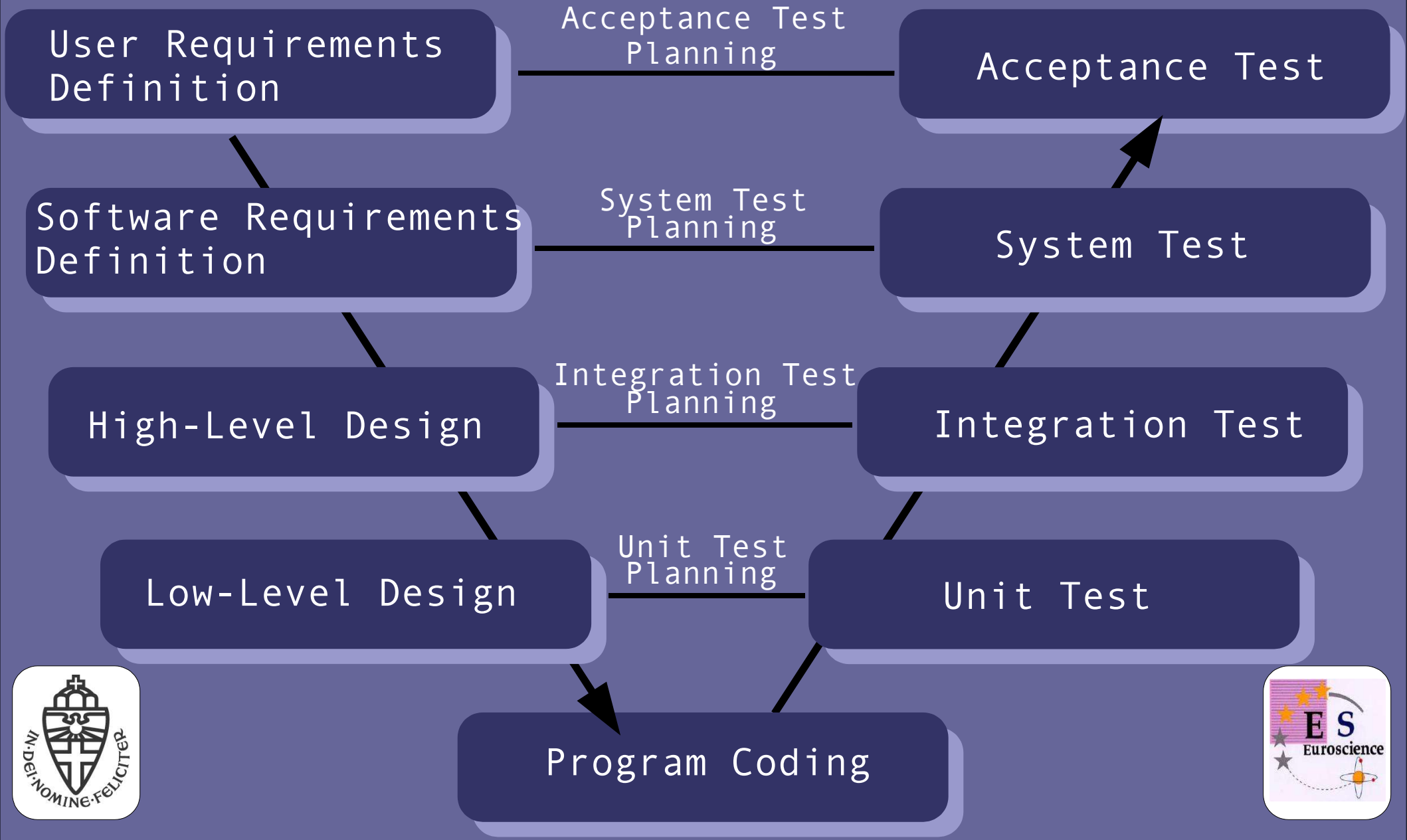
Testing - Kinds

- Static Testing:
- Structured Group Testing
  - Reviews
- Static Analysis
  - Compiler
  - Standards
  - Data Flow Analysis
  - Control Flow Analysis
  - Metrices

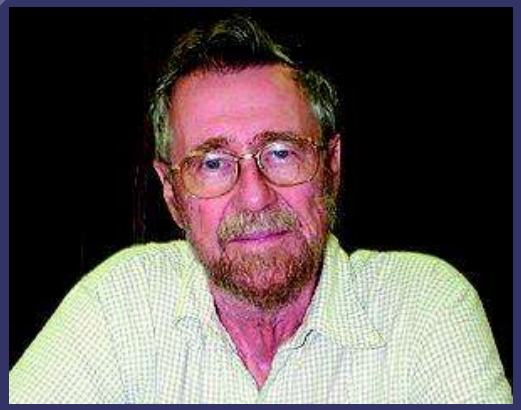
- Dynamic Testing:
- Blackbox Testing
  - Equivalence Partitioning
  - State-Based Testing
- Whitebox Testing
  - Instruction Coverage
  - Branch Coverage
  - Condition Coverage
  - Path Coverage

Intuitive Test Case Investigation!

The (advanced) V-Process



The Testing Limits



Testing can never demonstrate the absence of errors, only their presence.

Edsger W. Dijkstra

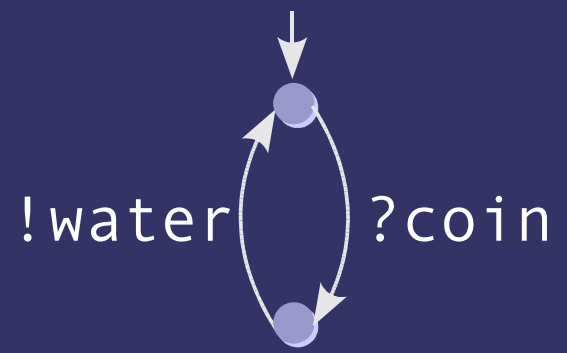
Systems are:

- Always a Black Box
  - → Experimenting is the solely approach to gather knowledge.
- Complex and Non-Terminating
  - → They elude exhaustive testing, i.e. proving.

→ Test Case Selection!

Reactive Systems - A Water Spender

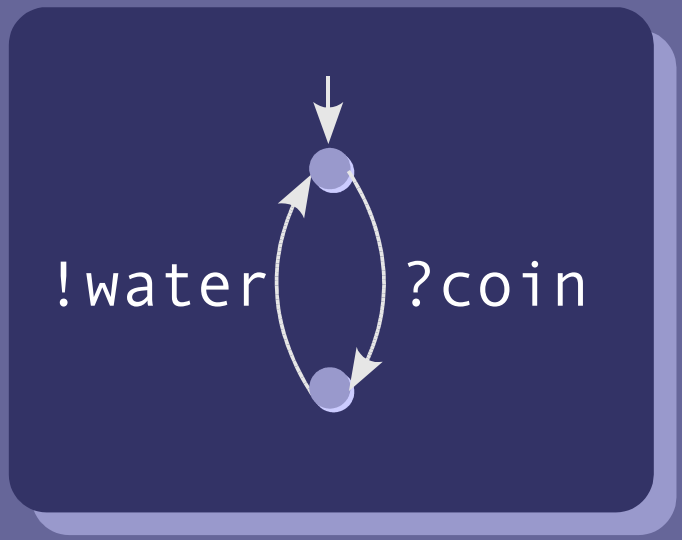
Model



System

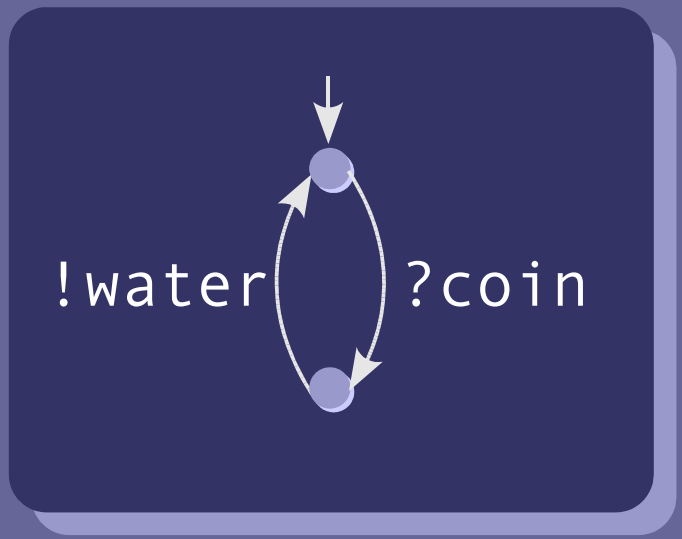
Test Hypothesis

Model ∋ System



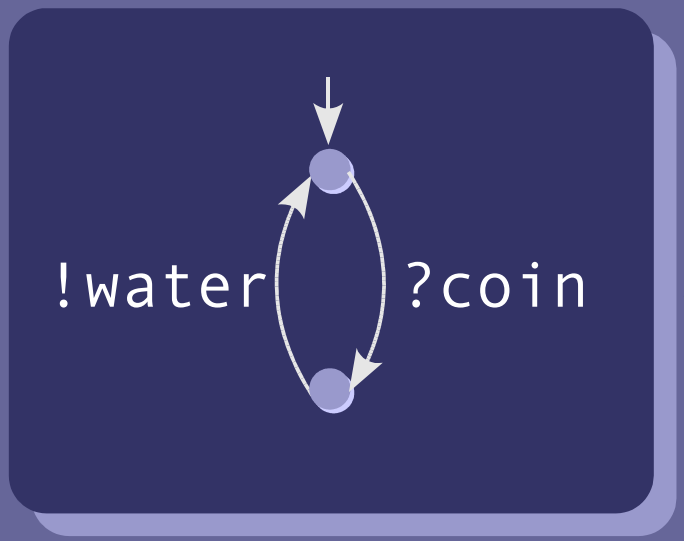
Model-Based Testing

Model ∋ System

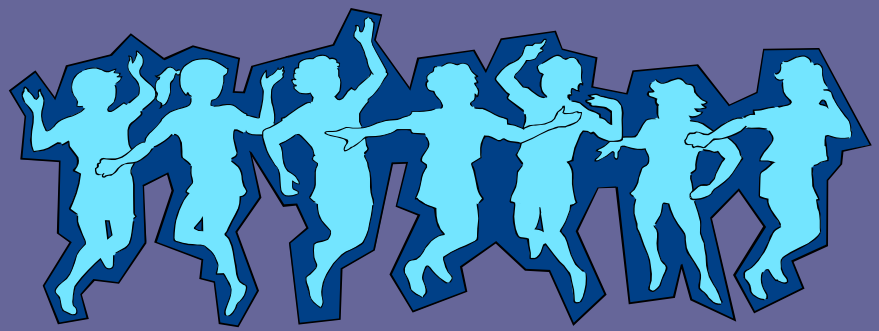


Model-Based Testing

Model ∋ System



conforms



IN·DEI·NOMINE·FELICITER

ES Euroscience

What is Conformance?

First approach: Equivalence!

Mealy Machine $\cong$ System

$M = \langle I, O, S, \delta, \lambda \rangle$ where

- I, O, and S are finite, nonempty sets of *input symbols*, *output symbols*, and *states* respectively,
- $\delta : S \times I \to S$ is the *state transition function*,
- $\lambda : S \times I \to O$ is the *output function*.

What is Conformance?

First approach: Equivalence!

Mealy Machine

?coin / !water

≁

Mealy Machine

Two machines are equivalent iff they produce the same output for every input sequence.

What is Conformance?

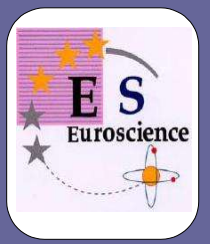
First approach: Equivalence!

Problem: Infinite input sequences!
The water spender has to fulfill: $?coin^n \rightarrow !water^n$

Dijkstra revisited:
Systems generally elude exhaustive testing!

"Solution" 1: reduce reality by assuming:
The system has no more states than the model...

"Solution" 2: give up equivalence!



What is Conformance?

Second approach: Weaker relations

LTS | conf | System

$\mathrm{M} = \langle S, L, T, s_0 \rangle$ where

- S is a countable, non-empty set of *states*,
- $L = L_I + L_U$ is a countable set of *labels*,
- $T \subseteq S \times (L \cup \{\tau\}) \times S$ is the *transition relation*,
- $s_0 \in S$ is the *initial state*.

What is Conformance?

Second approach: Weaker relations

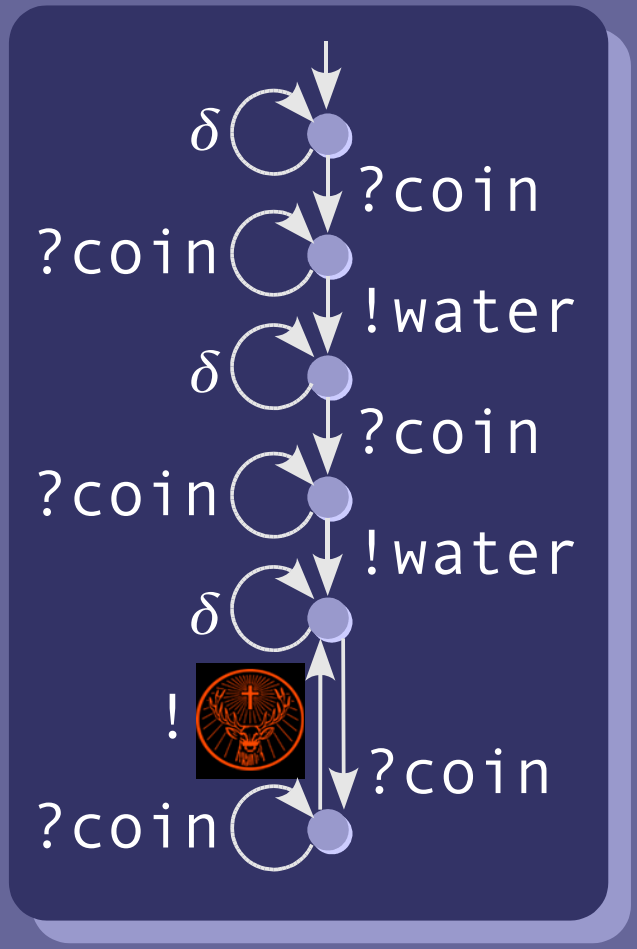
Labelled Transition Systems – Characteristics

- Potentially infinite state space
- Potentially nondeterministic
- Compositional
- Asynchronous I/O behaviour
- Notion of quiescence

What is Conformance?

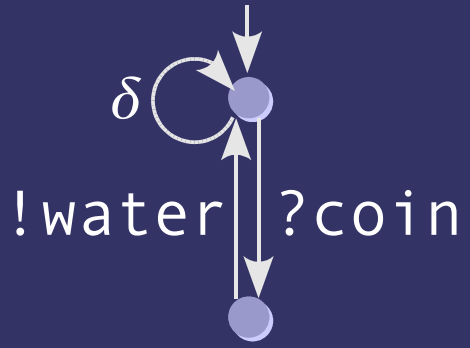
The ioco relation

Systems are assumed to be an input-enabled LTS (IOTS)

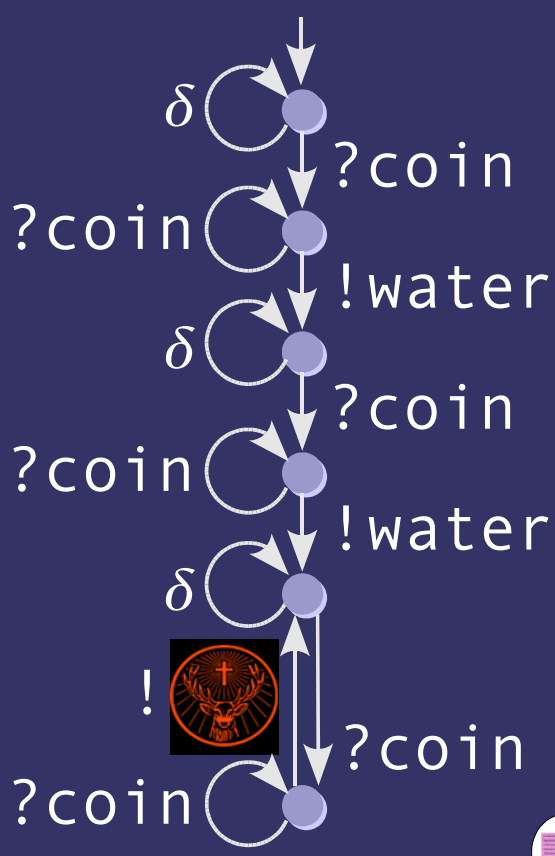


What is Conformance?

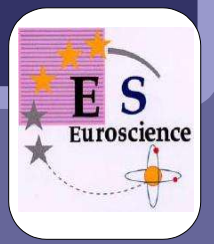
The ioco relation



ioco

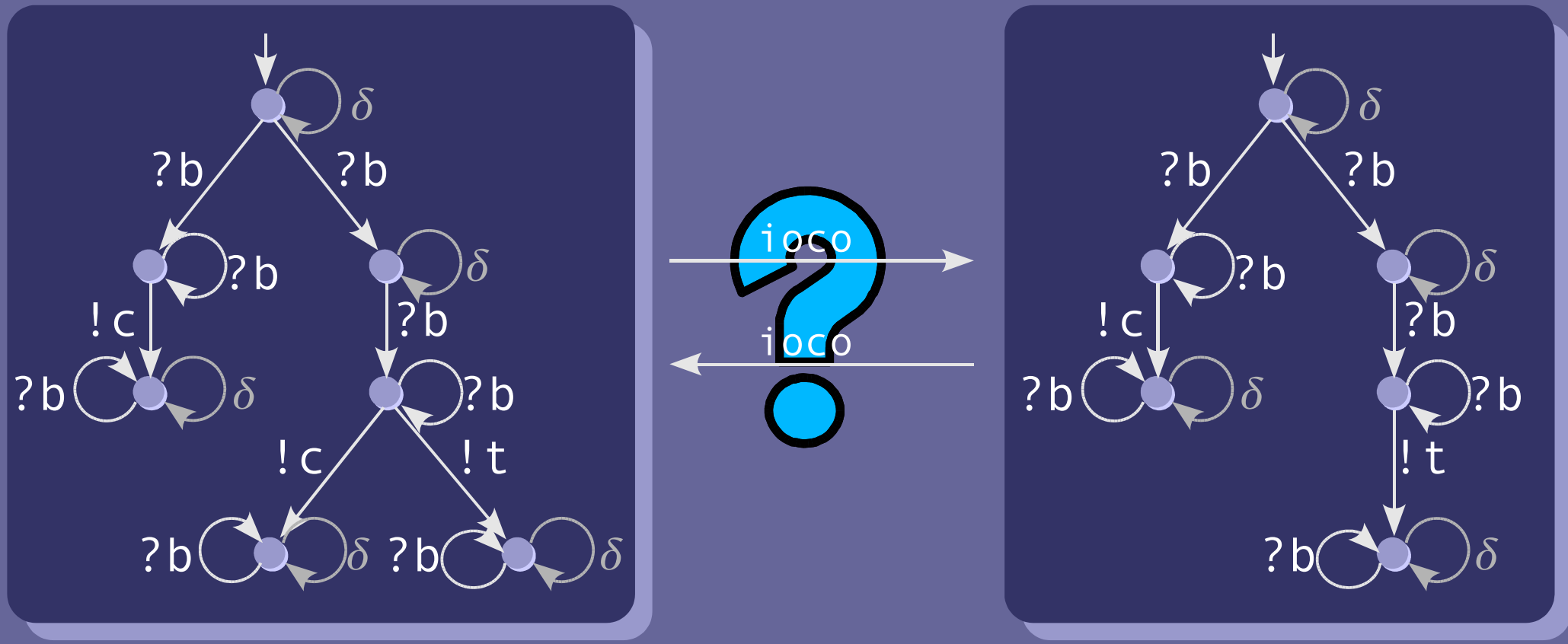


$$i \text{ ioco } s \Leftrightarrow_{def}$$
$$\forall \sigma \in \text{Straces}(s) : \text{ out}(i \text{ after } \sigma) \subseteq \text{out}(s \text{ after } \sigma)$$



What is Conformance?

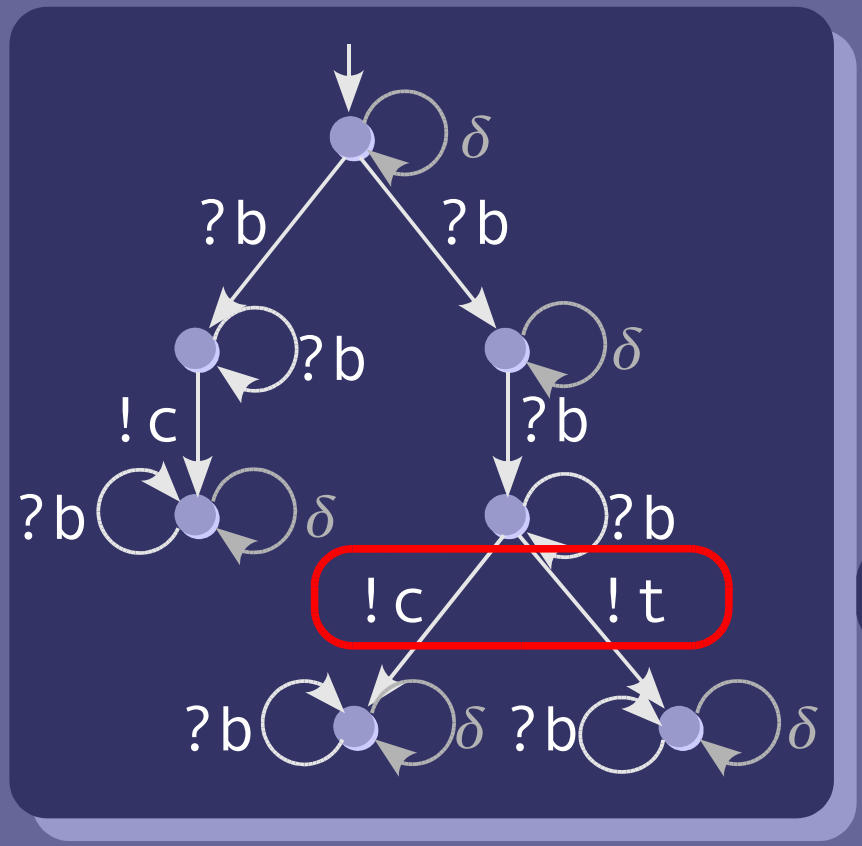
The ioco relation



$$i \text{ ioco } s \Leftrightarrow_{def}$$
$$\forall \sigma \in \text{Straces}(s) : \ \text{out}(i \text{ after } \sigma) \subseteq \text{out}(s \text{ after } \sigma)$$

What is Conformance?

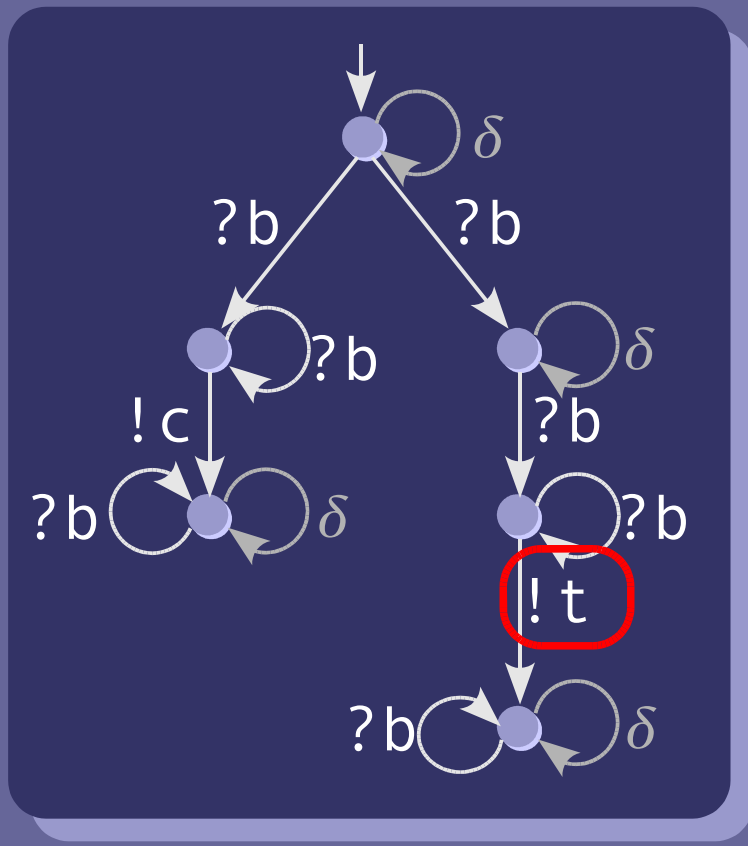
The ioco relation



ioco

ioco

out(?b δ ?b)



$$i \text{ ioco } s \Leftrightarrow_{def}$$
$$\forall \sigma \in \text{Straces}(s) : \text{ out}(i \text{ after } \sigma) \subseteq \text{out}(s \text{ after } \sigma)$$

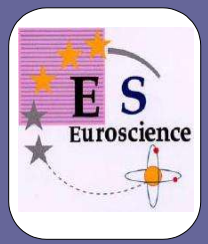
What is Conformance?

The ioco relation

Let s be a specification, and T a test suite, then

$$
\begin{array}{lllllll}
T \text{ is complete} & \Leftrightarrow_{def} & \forall i: & i \textbf{ ioco } s & \Leftrightarrow & i \textbf{ passes } T \\
T \text{ is sound} & \Leftrightarrow_{def} & \forall i: & i \textbf{ ioco } s & \Rightarrow & i \textbf{ passes } T \\
T \text{ is exhaustive} & \Leftrightarrow_{def} & \forall i: & i \textbf{ ioco } s & \Leftarrow & i \textbf{ passes } T
\end{array}
$$

A simple algorithm can be given which generates testcases, converging to a complete test suite.

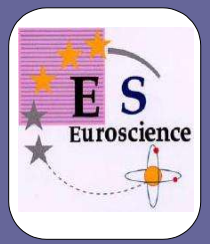


The TorX Tool

The ioco theory was implemented in the TorX tool:
http://www.purl.org/net/torx/

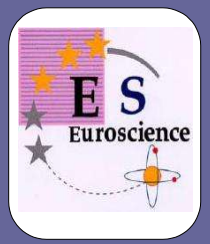
Within the STRESS-project, the ioco theory is extended w.r.t.

- Symbolic-data testing
- Real-time testing



Literature

- Broy, Jonsson, Katoen, Leucker, Pretschner (Eds.):
  Model-based Testing of Reactive Systems - A seminar volume
  LNCS, to appear in 2004
- Brinksma, Tretmans:
  Testing Transition Systems: An Annotated Bibliography
  LNCS 2067:187-195, 2001
- Lee, Yannakakis:
  Principles and Methods of Testing Finite State Machines - a Survey
  Proc. IEEE, 48(8):1090-1126, 1996



- Thank You!