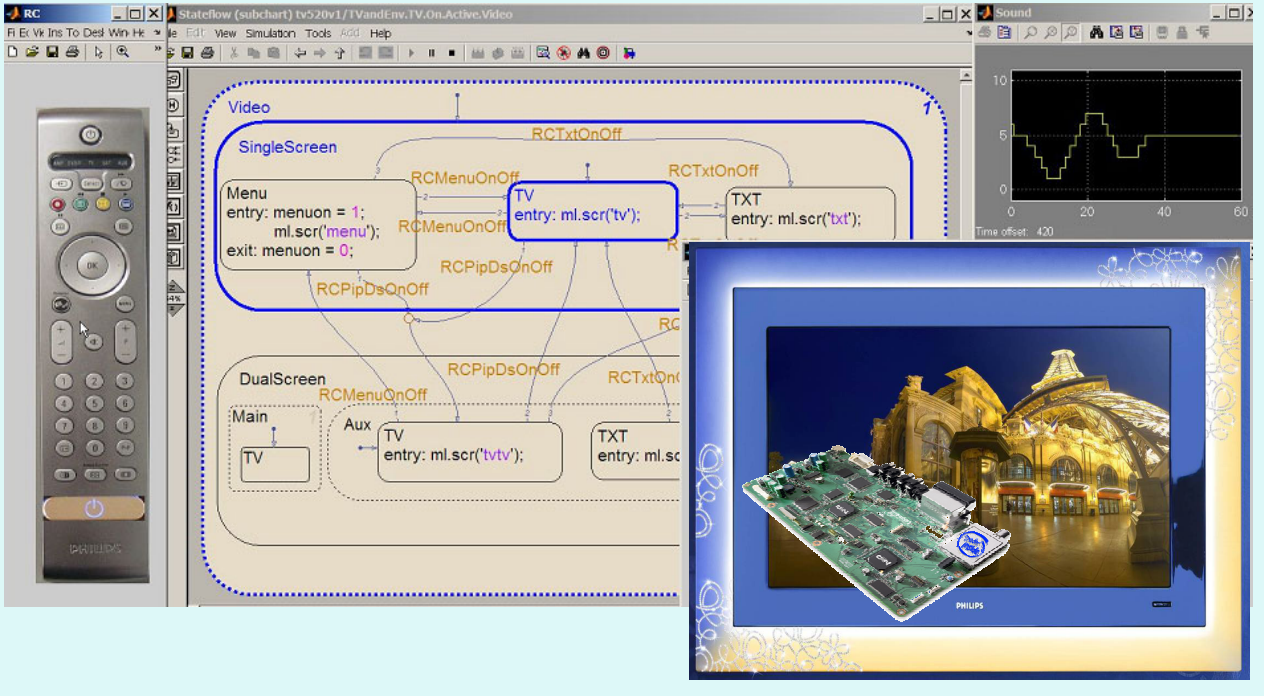
Trader
Model-based Awareness to Improve Reliability

Jozef Hooman, ESI

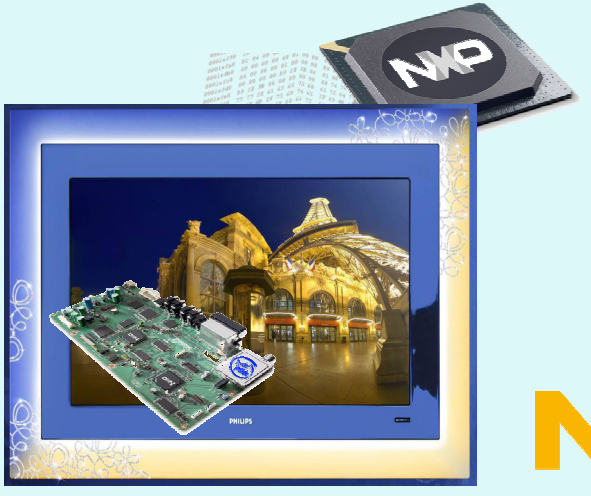


Outline

- **Overview of the Trader Project**
- **Modeling the TV behaviour – demo**
- **Using TV models for awareness**
- **Modeling awareness – demo**
- **Concluding remarks**

TRADER
System Reliability

Embedded Systems INSTITUTE



Period: Sept. 2004 - Aug. 2009
20 fte/yr, 7 PhDs,
1 Postdoc, 10 Partners

Carrying Industrial Partner

Goal

Develop methods and tools to optimize reliability of high-volume products.

Issues

- Minimize product failures.
- Increase user satisfaction (user-centric design approach)

Research

Embedded Systems INSTITUTE

Reliability threats

- **TV Complexity increase follows the PC industry**
  - **Functions/content increases rapidly**
    - *SW content increases rapidly*
    - *Third party content increases (EPG, codec's)*
  - **External information sources multiply**
    - *Connected planet strategy*
    - *Downloadable applications*
- **Customer's TV reliability expectations are not lowered**
- **Reliability must not hamper time-to-market**
  - **Fixed shipping gates to occupy reserved shelf space**

➔ **Fault in delivered products are unavoidable**

TRADER

Trader research challenge

Prevent product faults causing customer complaints

- **How do customers perceive/rate reliability for high-volume products?**
- **What are reliability options applicable for such highly cost sensitive products?**
- **How can reliability be made a design goal of such a product rather than a 'hit or miss' property?**

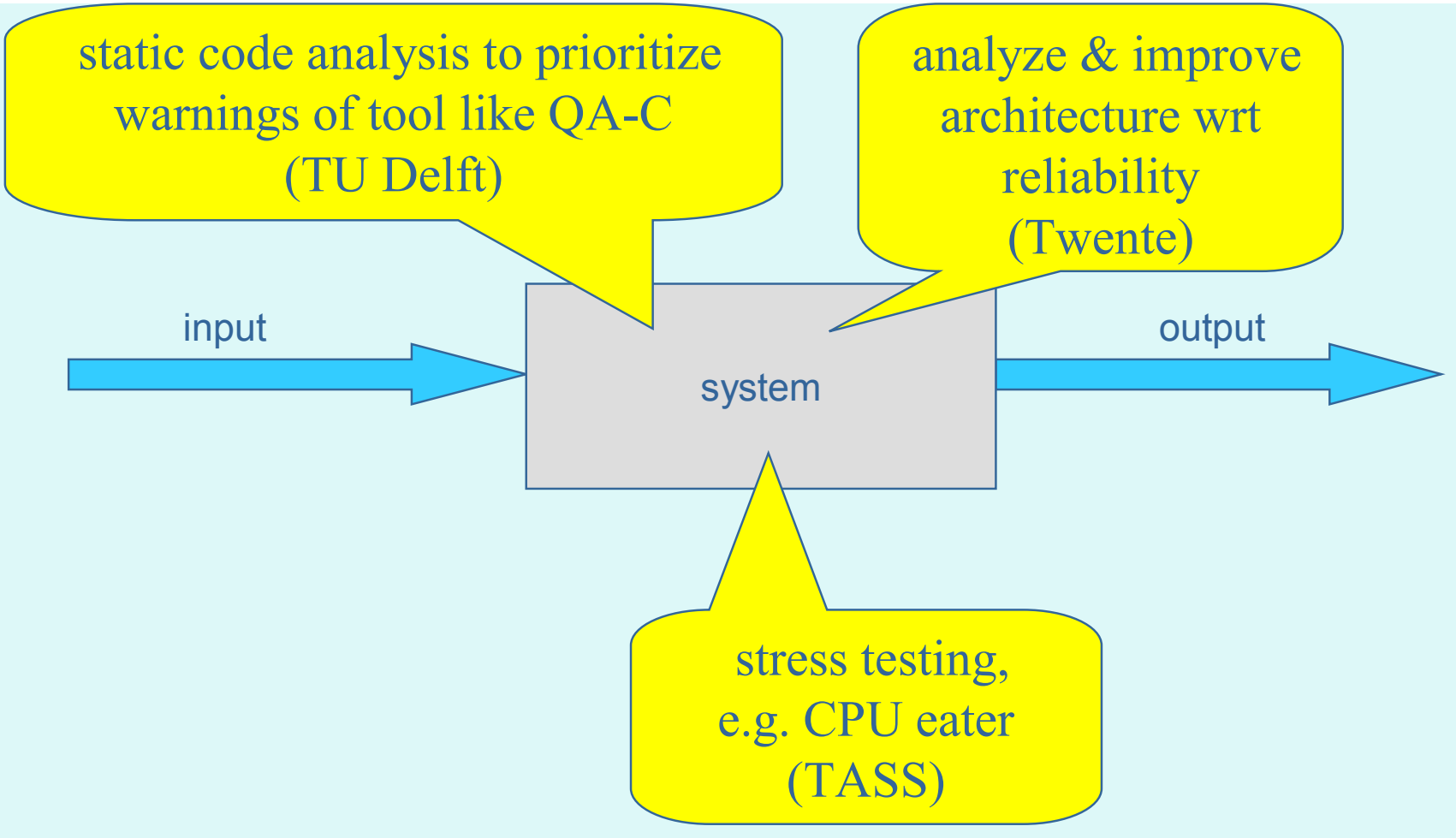
Trader – Proposed Outcome

Embedded Systems INSTITUTE

- Methods and techniques that
  - **can expose,** *at design time*, **product** *weaknesses* **that could lead to erroneous behavior**
  - **give the system** *awareness* **that its** *customer-perceived behavior* **is or is likely to become erroneous**
  - **provide the system with a** *strategy* **to** *correct* **itself in line with customer expectations**
- **Supported by**
  - *Proof of concepts* **&** *publications* **that show the "how"**
  - *Knowledge transfer* **to CIP and industry**

TRADER

Realizing the Trader vision



Trader – Proposed Outcome

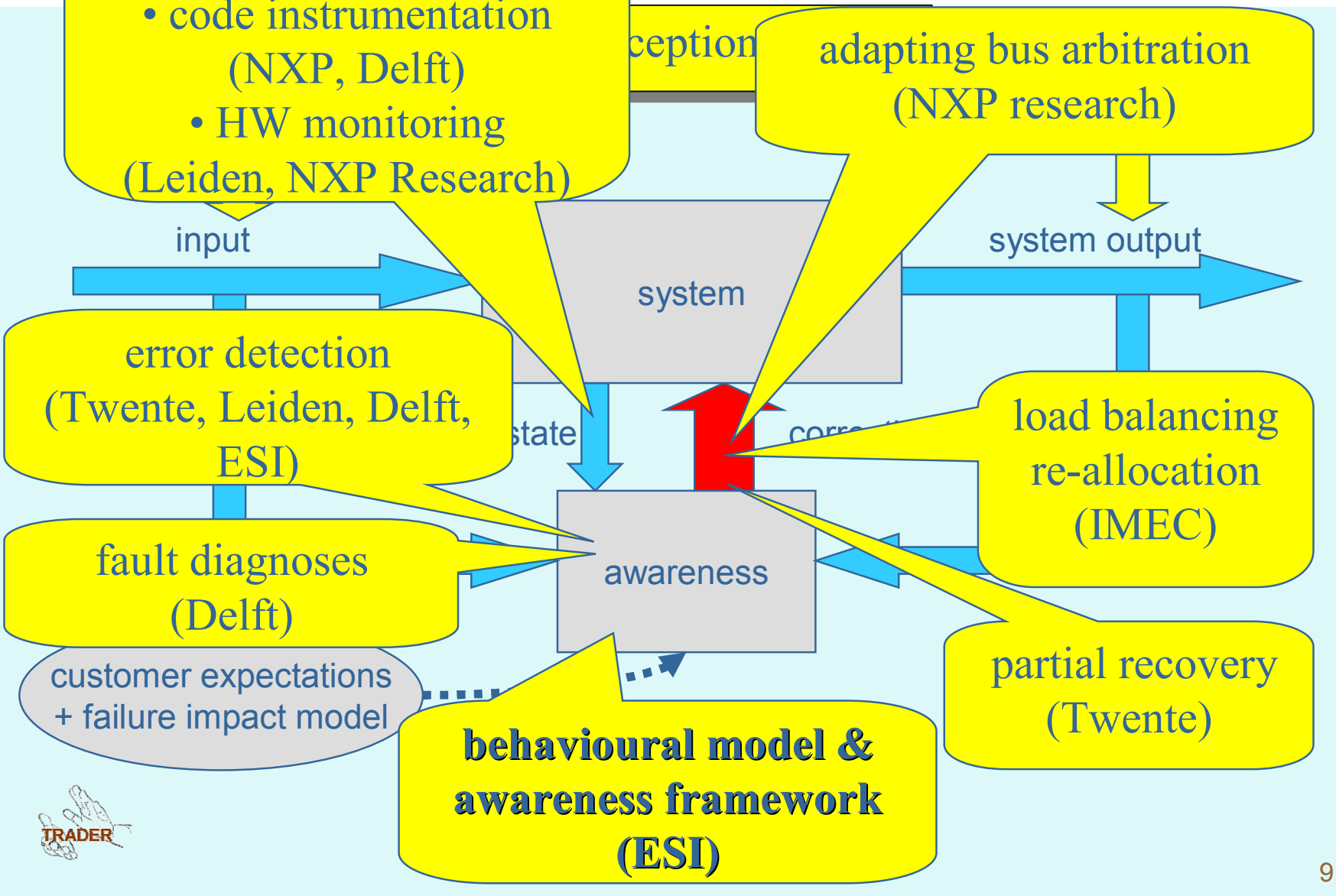
Embedded Systems INSTITUTE

- Methods and techniques that
  - **can expose,** *at design time*, **product** *weaknesses* **that could lead to erroneous behavior**
  - **give the system** *awareness* **that its** *customer-perceived behavior* **is or is likely to become erroneous**
  - **provide the system with a** *strategy* **to** *correct* **itself in line with customer expectations**
- **Supported by**
  - *Proof of concepts* **&** *publications* **that show the "how"**
  - *Knowledge transfer* **to CIP and industry**

TRADER

R vision

Embedded Systems INSTITUTE



TRADER

Focus of current TV model

Embedded Systems INSTITUTE

Current modeling work concentrates on

- **global control of TV**
- **user perceived behaviour;**
  interaction via Remote Control and TV buttons

Ignore

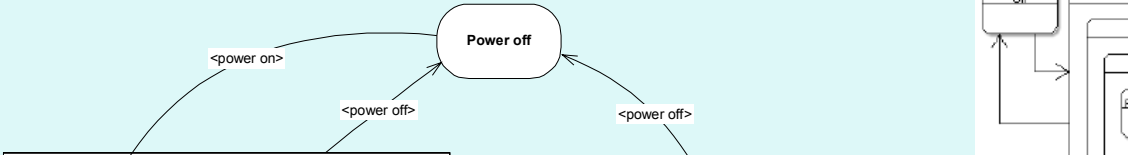
- **installation, settings**
- **image/sound quality**
- **screen size, view modes**
- **ambient light**
- **external devices**
- **internal modes & components**
- **…**

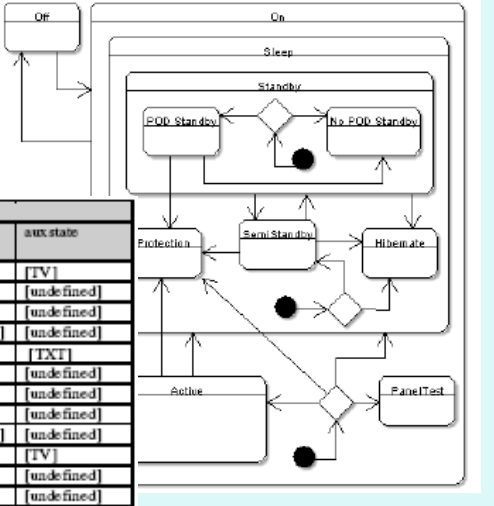
TRADER

What is a model?

Embedded Systems INSTITUTE

Documents contain lists of requirements, transition tables and pictures, especially state diagrams





			Event	End	
		auxstate		active screen	mainstate	auxstate
		[undefined]	RCPipDsO	[main]	[TV]	[TV]
			RCTxtOnOff	[main]	[TXT]	[undefined]
			Menu On	[main]	[Menu]	[undefined]
			USB break-in	[main]	[Content Browser]	[undefined]
[main]	[TXT]	[undefined]	RCPipDsOnOff (2)	[aux]	[TV]	[TXT]
			RCTxtOnOff	[main]	[TV]	[undefined]
			[SleeptimerLastMinute]	[main]	[TV]	[undefined]
			<txt off> [1]	[main]	[TV]	[undefined]
			USB break-in	[main]	[Content Browser]	[undefined]
[main]	[Menu]	[undefined]	RCPipDsOnOff [10] (2)	[main]	[TV]	[TV]
			RCTxtOnOff (1)	[main]	[TXT]	[undefined]
			Menu Off	[main]	[TV]	[undefined]
			StartingDemo	[main]	[TV]	[undefined]
			USB break-in	[main]	[Content Browser]	[undefined]
[main] / [aux]	[TV]	[TV]	RCPipDsOnOff	[main]	[TV]	[undefined]
			RCTxtOnOff (1)	[main]	[TXT] [V] -- CR 3114 --	[undefined]
			Menu On	[main]	[Menu]	[undefined]
			USB break-in	[main]	[Content Browser]	[undefined]
[aux]	[TV]	[TXT]	RCPipDsOnOff	[main]	[TXT]	[undefined]

Conclusion: model control behaviour using state diagrams

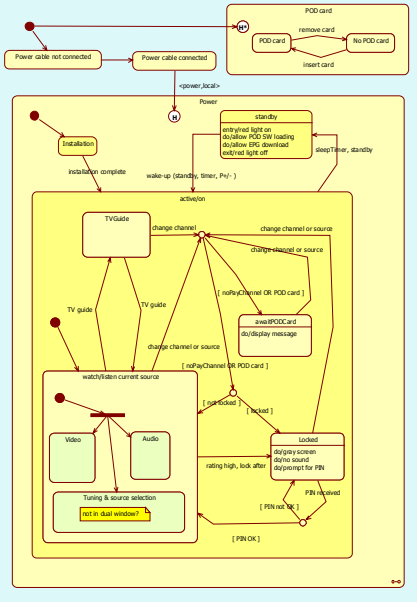
To deal with complexity we need hierarchy, concurrency, …

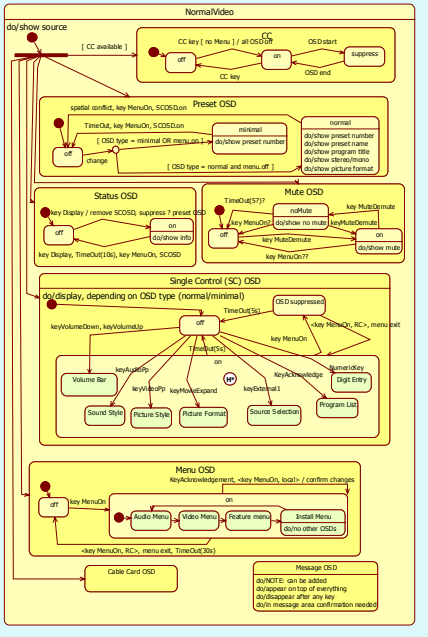
TRADER

Experiment with modeling

Embedded Systems INSTITUTE

Used free UML tool (starUML) to express high-level model, based on NXP documents, user manuals, TV experiments





Conclusion: feasible, but

- **Difficult to get required documentation & information**
- **Easy to make modeling errors, difficult to find them**

Needed: executable model

TRADER

Model TV behaviour

Aim: model user perceived behaviour of TV mainly by executable state diagrams and visualize external IO

Current tool support: Matlab/Simulink, mainly using Stateflow toolbox to define executable state diagrams with hierarchy, concurrency, data, events, …

Approach is rather tool-independent; diagrams are similar to state machines in UML-tools such as Rose RealTime and Rhapsody

DEMO

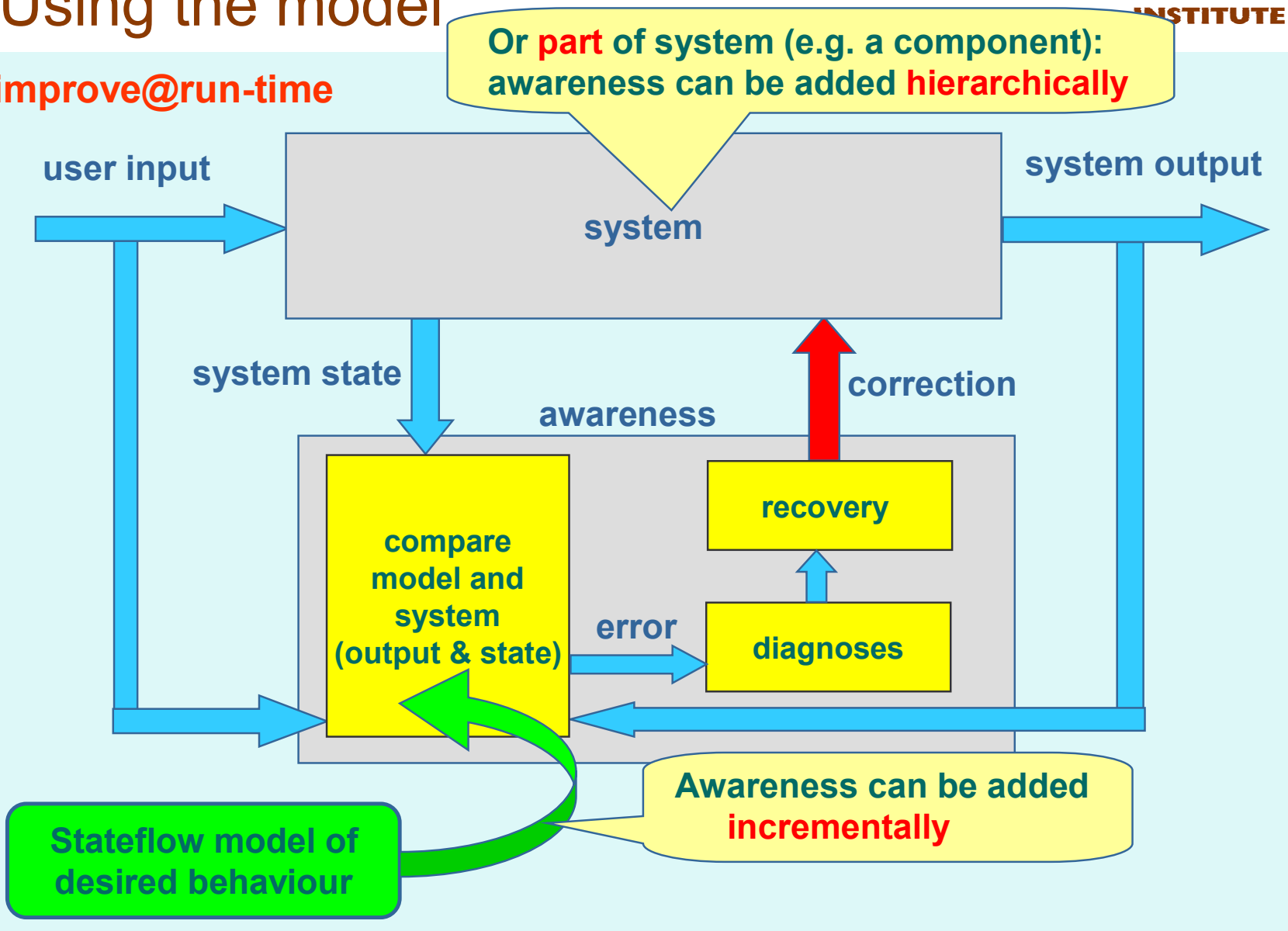
Using the TV model (1)

improve@development:

- to obtain concise, visual specification;
  currently spec is distributed over many documents
- to enable early detection of faults
  (e.g. ambiguities, omissions, inconsistencies, interference between features)
- to get quick feedback on product variations
- to generate test cases to check conformance of implementations w.r.t. specification

Using the model

improve@run-time



Using the TV model (2)

improve@run-time

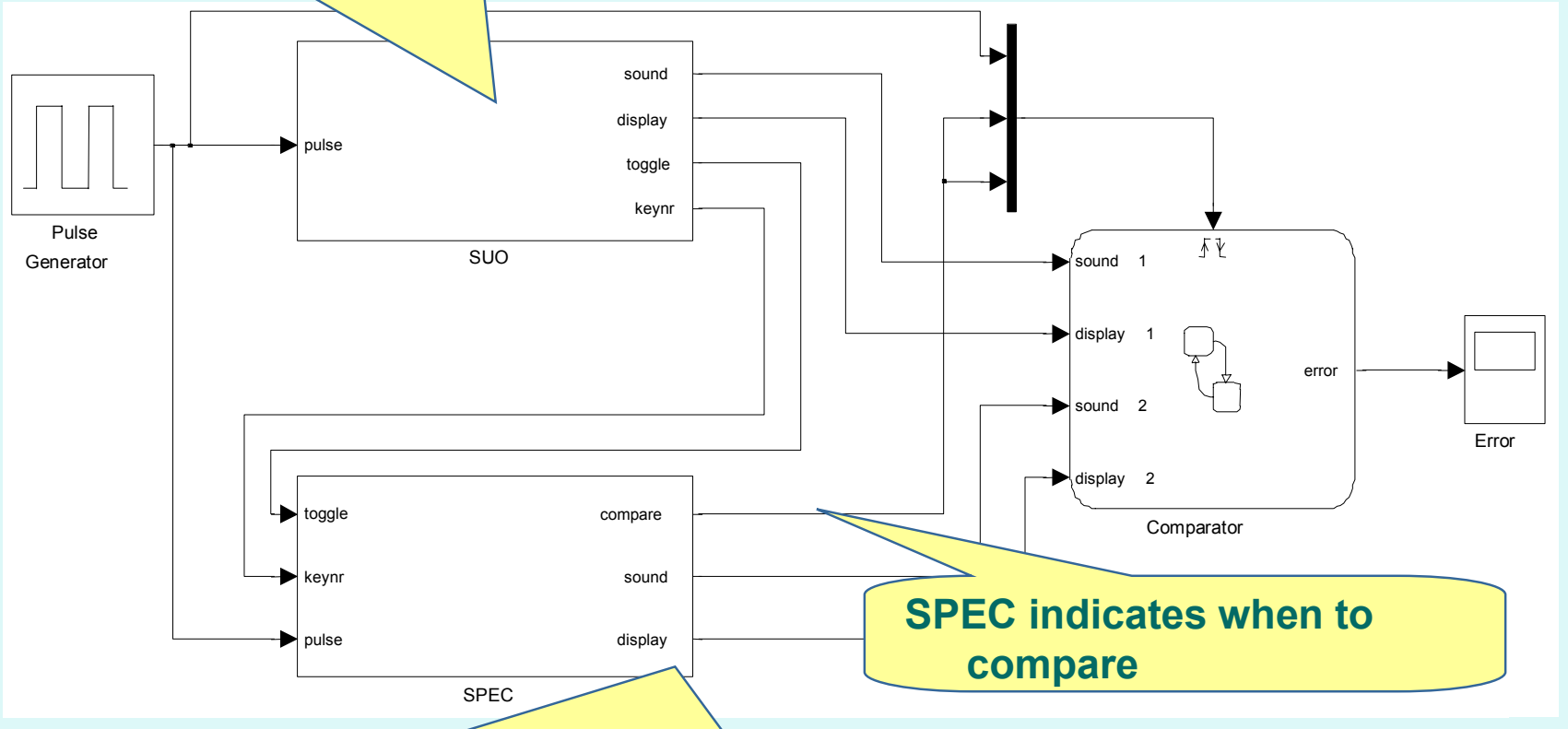
Current work: experiment with awareness concept

- **Linux-based awareness framework in which System Under Observation (SUO) and SPEC can be inserted easily and we can try different error detection strategies**
- **Open source media player MPlayer as first case study, followed by experiments in TV domain**
- **Model awareness concepts in Stateflow**

Awareness in Stateflow

Embedded Systems INSTITUTE

Input events are queued and when processed also sent to SPEC



SPEC indicates when to compare

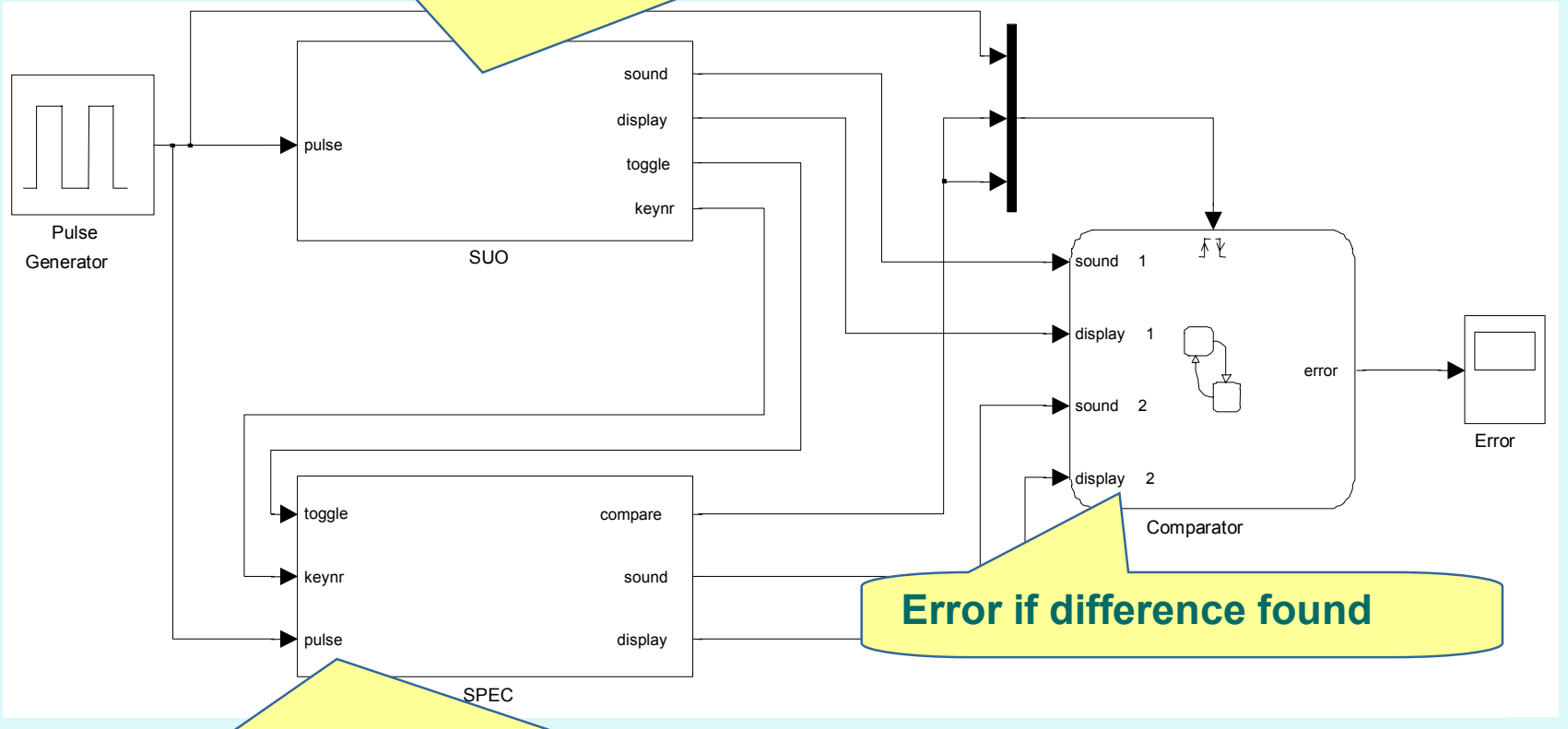
Specifies worst-case timing, delays are interrupted by new events from SUO

TRADER

Example of sound and display

Embedded Systems INSTITUTE

Volume delay: 1 Mute delay: 1 Unmute delay: 2
Display time: 5



Error if difference found

Volume delay <= 2 Mute delay <= 2 Unmute delay in [1,3]
Unmute display time in [3,7]

DEMO

Concluding remarks (1)

First Trader results applied at design time, for instance:

- **Stress testing, CPU eater already included in current release and can be activated in request**
- **INXS for aspect-oriented observations, e.g. to increase insight in platform usage**
- **Spectra-based diagnoses to support debugging**

Concluding remarks (2)

Embedded Systems INSTITUTE

Also, TV model useful at design time

- **to detect problems early (e.g. by visualization of IO)**
- **to derive test cases**

In general, high-level system model (from user point of view) is missing

Current focus:

- **Use of the model to create awareness**
- **Use of the model for requirements capture and testing**
- **Include video and e.g. view modes (4:3, 16:9, ..)**

Thank you for your attention!

TRADER