# TTC 2011 Live Contest

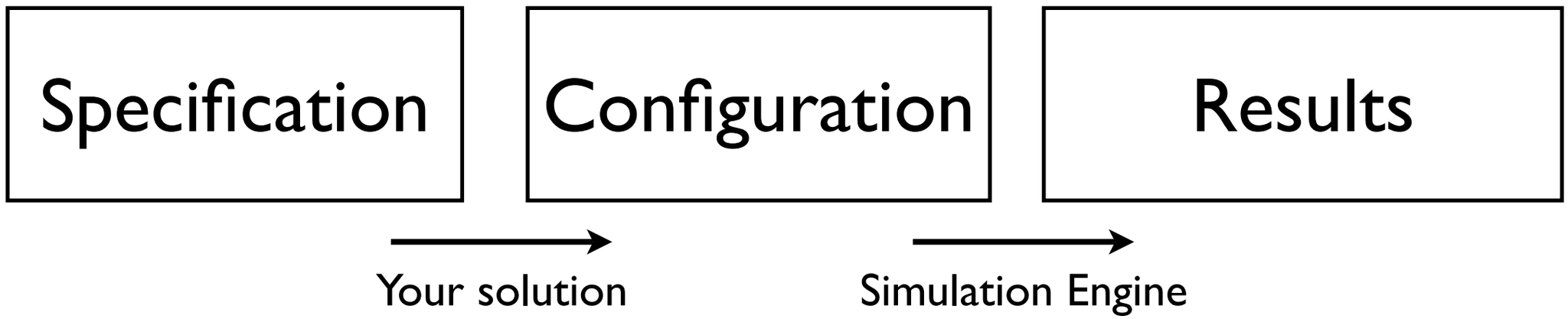
Louis Rose
THE UNIVERSITY of York

# Themes

- Transformation for interoperability
- Behavioural modelling
- Model matching
- (A little) model-based testing

# Overview

**Metamodels:**




**Core Task**:

Transform specifications to configurations.

# Task Resources

- All resources are stored on GitHub: https://github.com/louismrose/ttc2011
  - Metamodels.
  - Source models.
  - Reference target models.
  - Instructions and these slides.

# Core Task

## Configuring the simulation

# The Simulation Engine

## Domain: digital watches


- http://ttcsim.appspot.com
- ***Configured*** with an EMF model
- Produces an EMF ***results*** model

# A Simple Specification

**Given** the watch is in mode “on”
**Then** the first button must be called “off”

**When** the watch enters mode "on"
**Then** the "display" must show "hello"

**Given** the watch is in mode "on"
**When** the first button is pressed
**Then** the watch must be in mode "off"

...

# A Simple Specification

...

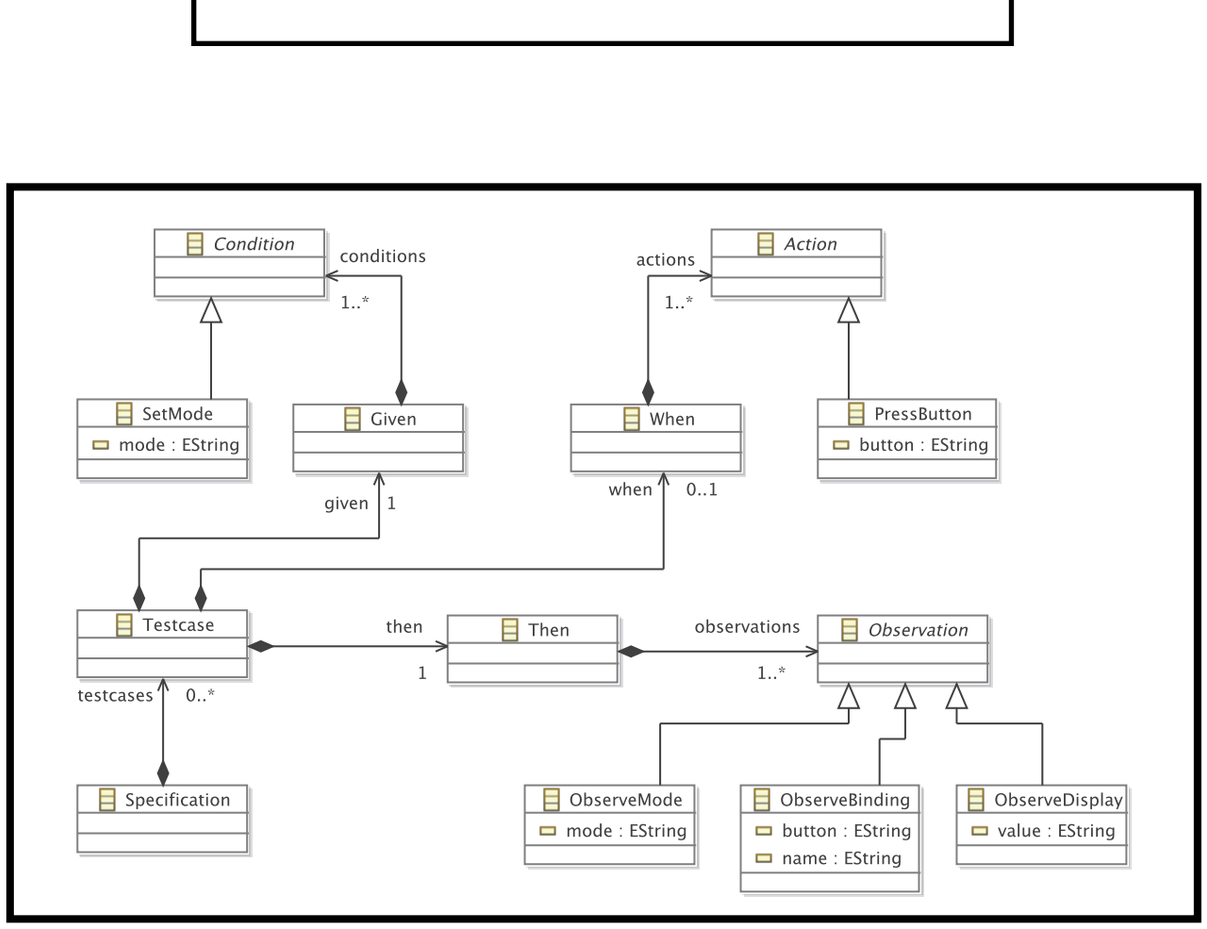
**Given** the watch is in mode “on”
**Then** the first button must be called “off”

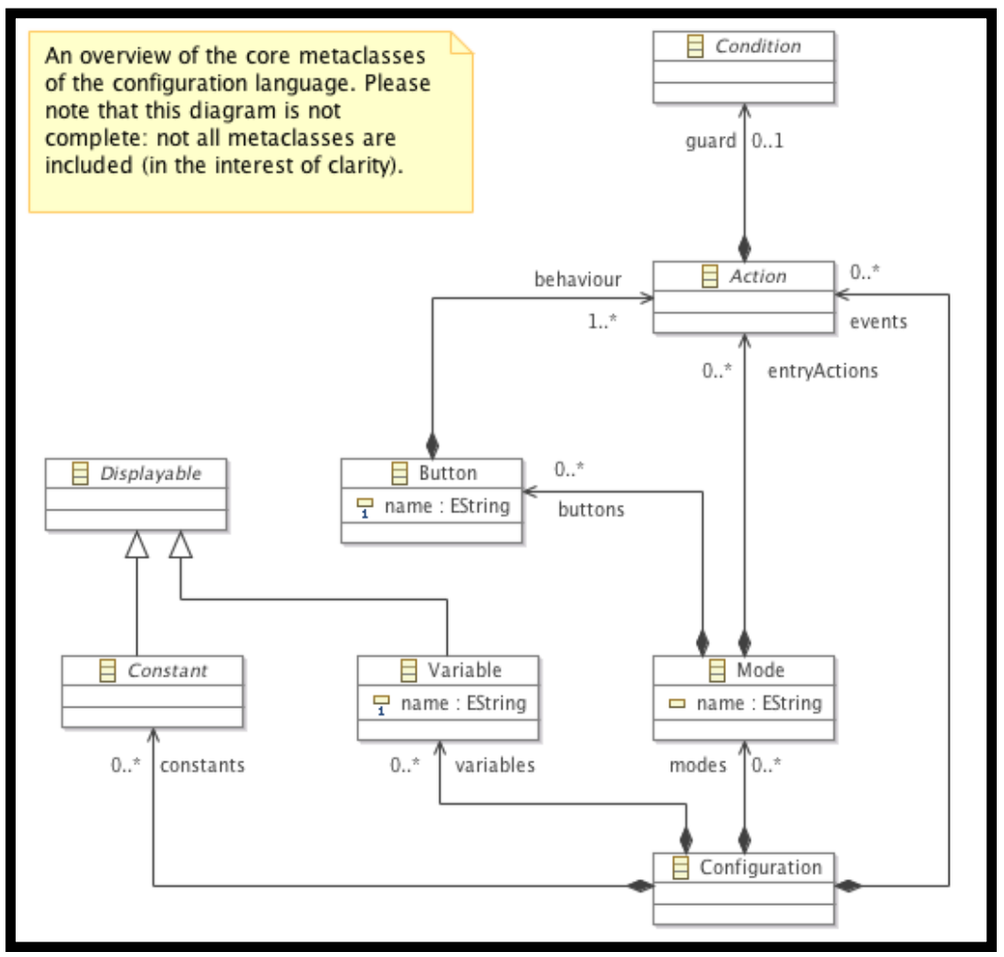
**When** the watch enters mode "on"
**Then** the "display" must show "hello"

**Given** the watch is in mode "on"
**When** the first button is pressed
**Then** the watch must be in mode "off"

# The Transformation

Specification → Simulation configuration

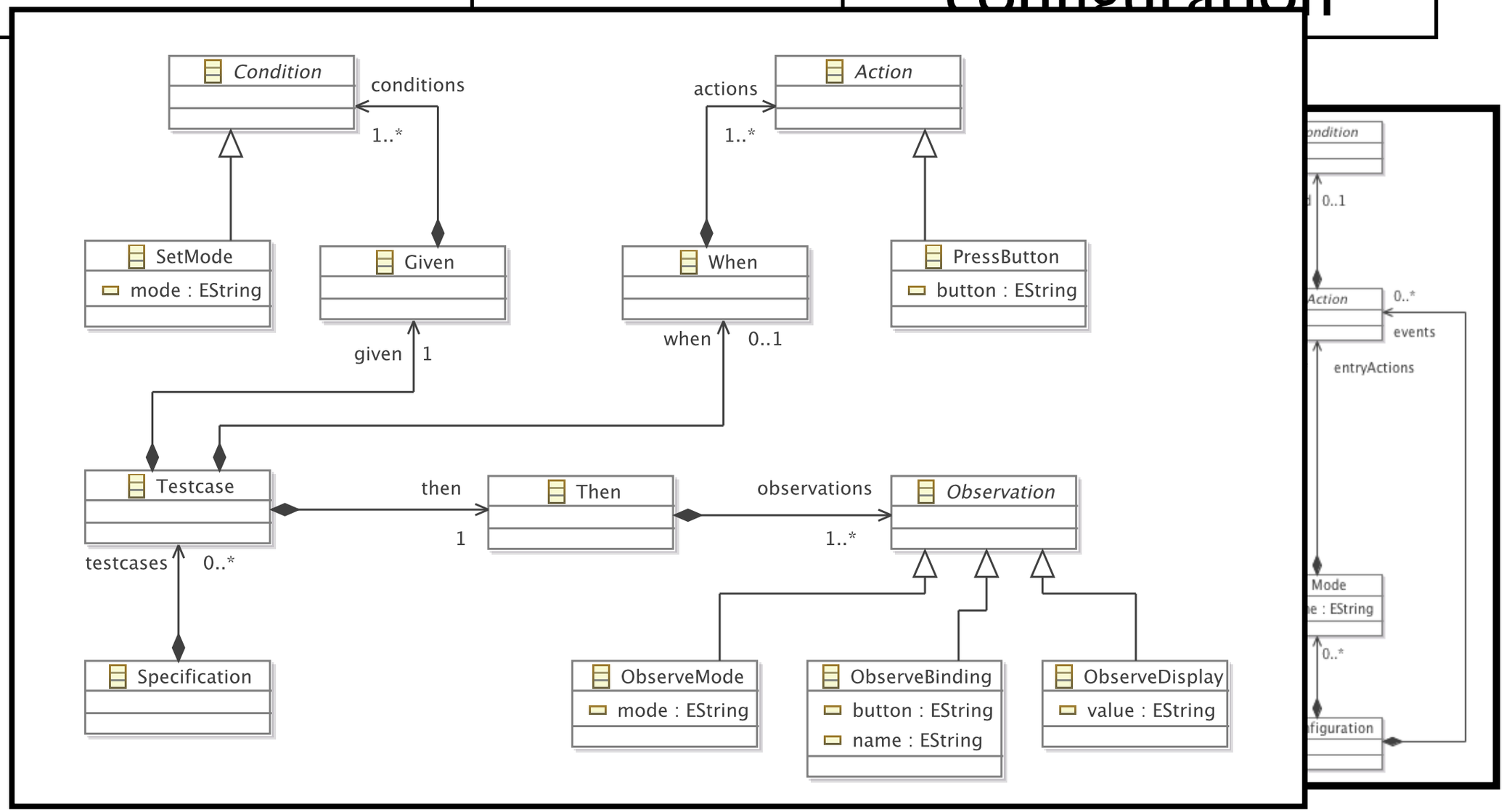


# The Transformation

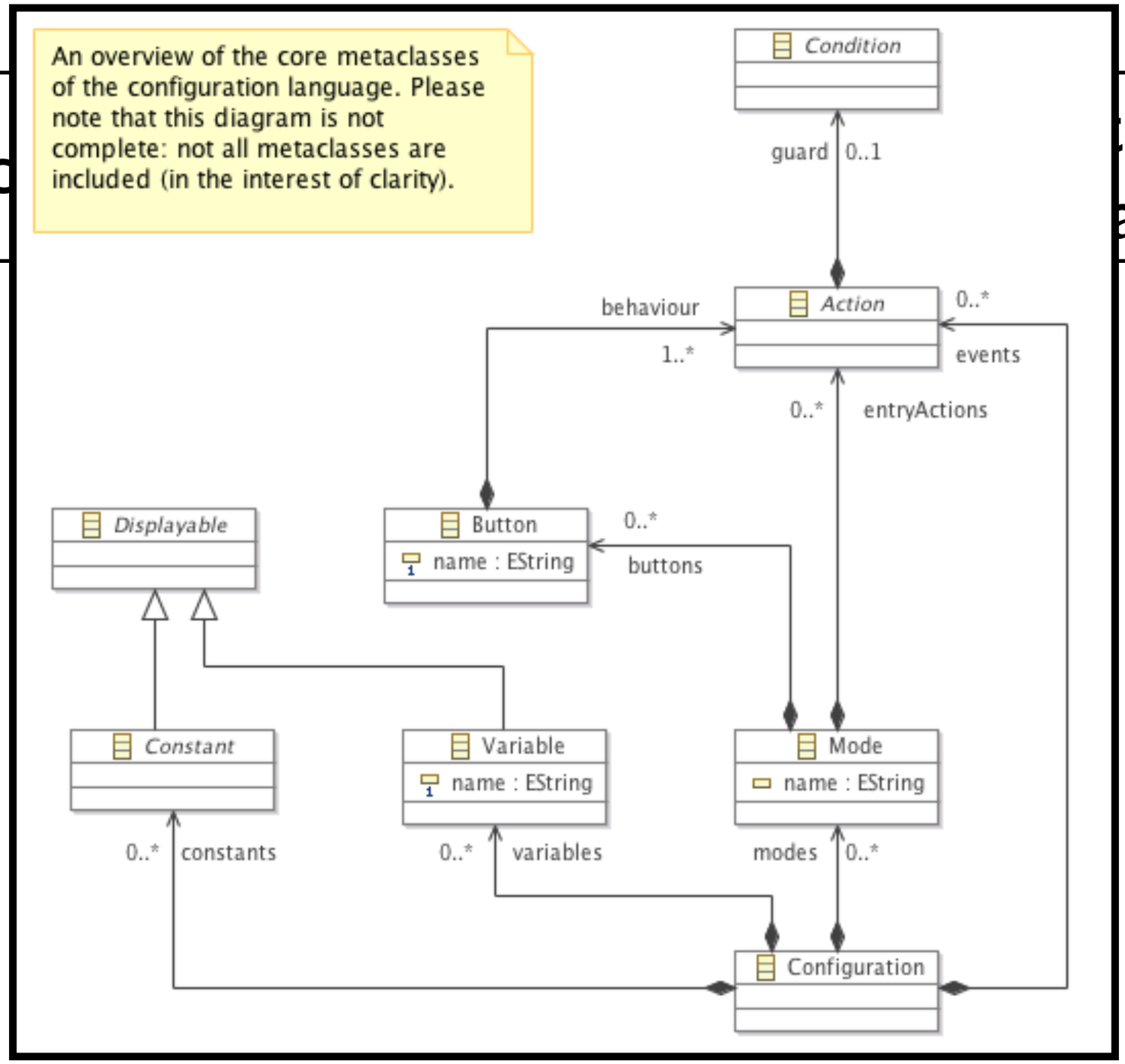
Specification

Simulation configuration



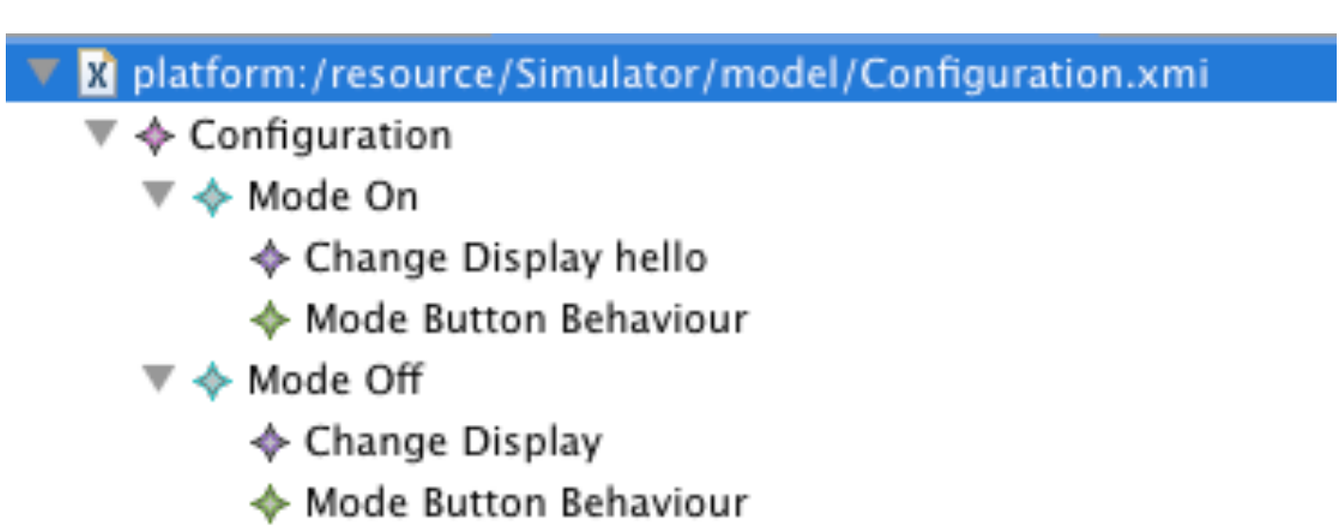
# The Transformation

Spec … ion … ation



# A Configuration

The on/off watch specification should be transformed to produce the following configuration:



# Extension 1

## Robustness of the transformation

# Unusual Specifications

**Given** the watch is in mode "alarmTime"
**Then** the third button must be called "minute"
the first button must be called "mode"
the second button must be called "hour"

**Given** the watch is in mode "time"
the "indicator" is showing "unset"
**When** the second button is pressed
**Then** the "indicator" must show "set"
the alarm must ring
the display must show "alarm"

# Unusual Specifications

- Can your transformation manage these unusual specifications?
- Can you define further types of (valid and consistent) specification that all solutions should aim to tolerate?

# Extension 2

## Matching test results with specifications

# Simulation Results

The simulation provides a results model:

hello

**Alarm:** not prepared
**Mode:** On

**Buttons**

Mode

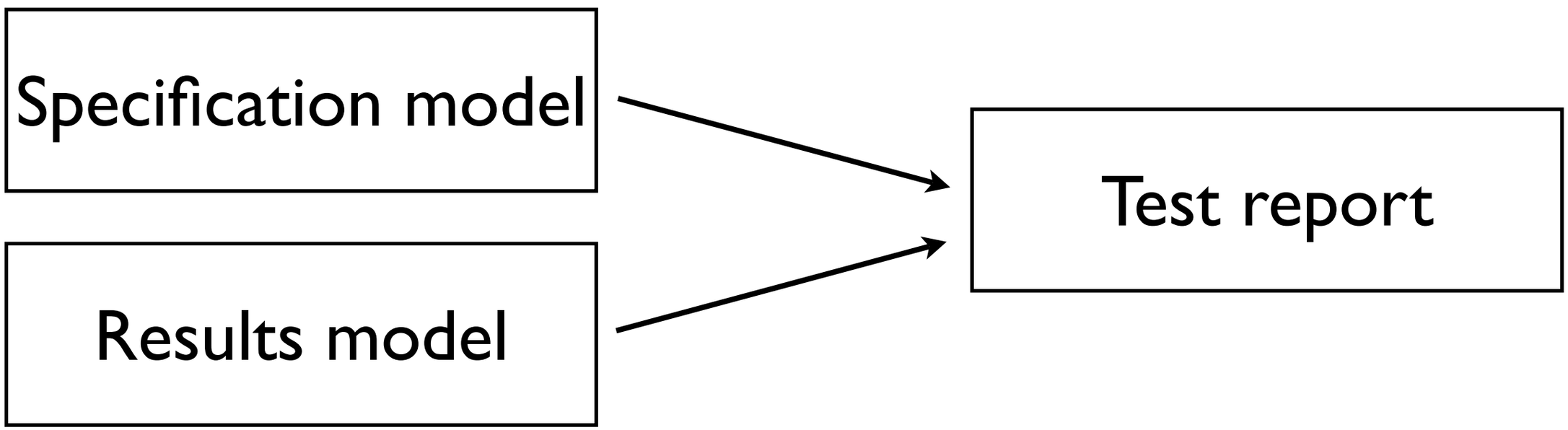
**Simulation Output**

- E type=mode, params=[On]
- E type=display, params=[hello]
- S type=button, params=[0]
- E type=mode, params=[Off]
- E type=display, params=[]
- S type=button, params=[0]

Download as EMF model

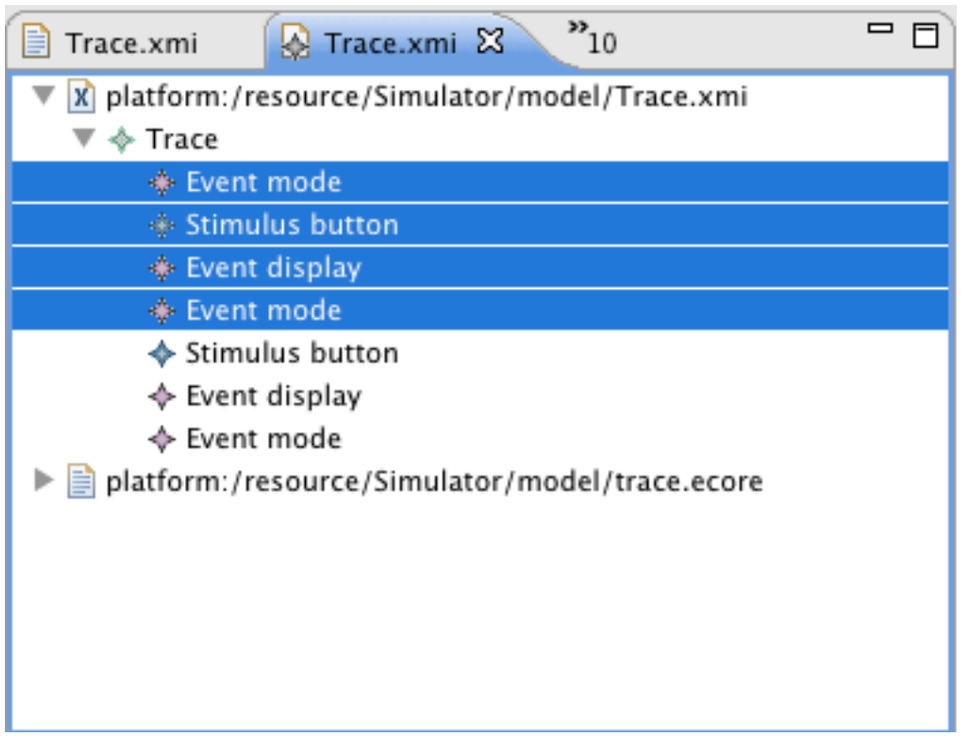
# Extension 2 (Matching)

Match the specification and results models to produce a test report



# Matching Example

**Given** the watch is in mode "off"
**When** the first button is pressed
**Then** the mode must be "on"
the display must show "hello"



# Matching Example

**Given** the watch is in mode “on”
**When** the first button is pressed
**Then** the mode must be “off”
the display must show “”