Application of Model-Based Testing to validation of new nuclear I&C architectures

François Chastrette, Frédérique Vallée – All4tec
Luc Coyette – Esterel Technologies
Contents

- Context
- Case study: a simple Instrumentation and Control (I&C) system
- I&C specification
- Model-Based Testing approach for early validation of an I&C specification
- Preliminary Results
- Conclusion
The Connexion Project

- A large cooperative R&D project
  - Goal: a modular workbench for functional engineering activities in nuclear I&C (Instrumentation and Control) systems
  - Includes all major French players in the nuclear I&C field
    - Operator + architect/engineer (EDF), I&C systems providers (AREVA, Rolls-Royce, Alstom, Atos), technology suppliers (Esterel Technologies, CORYS, All4tec, Predict), research labs (CEA, ENS Cachan, CRAN, Telecom Paris Tech, INRIA, INP Grenoble)
  - Targets improvement of tools
    - SCADE, MaTeLo, Safety Architect, CORYS ALICES, etc.

- This presentation covers only one sub-project
  - Early functional validation (i.e. validation of requirements before implementation)
    - Tests should be reusable throughout the system development lifecycle
  - Scope: one I&C function
    - I.e. the result of upstream analysis
Design phase

250 Elementary Systems (SE’s)

Source: EDF
Context – Lifecycle Overview – 2 / 2

One I&C Function

Source: EDF
Context – Lifecycle of an I&C Function

Functional requirements

1. Early functional validation

2. Functional requirements

Functional specification

3. Functional specification

4. Design specification

5. Design specification

6. Design specification

7. Implementation

8. Implementation

9. Implementation

10. Implementation

Early functional validation:

- SCADE
- MaTeLo

Text / SysML
Scade
Func. Diagrams
PLC

www.all4tec.net
Case study – Diesel Load Sequencer

- One particular I&C function
  - One of the components of a larger plant function: back-up power supply (BPS)
  - Is the control system for the Emergency Diesel Generator (EDG)

- Purpose: if Main Power Supply is lost, start the Diesel generators, then reconnect the major components of the plant
  - Provide an orderly load sequence to prevent heavy transients on the emergency diesel generator solicitation

- Features
  - Many Boolean inputs (24)
    - Input redundancy (AND / OR) is defined by Quality of Service requirements allocated to this function
  - Boolean outputs (12)
  - Complex timing constraints (orderly reload sequence: 7 or 9 scheduled actions at timed intervals)
Goal: Early validation of functional requirements

- Combination of Model Based Design and Model Based Testing
- Models developed by independent teams from same requirements

Model Based Design (MBD)

- Graphical model to describe a system from textual requirements
- Executable model allowing clarification of the early set of requirements and early testing (before writing any code)
- Full traceability to textual requirements
- All verifications performed at model level

The SCADE formal notation
Functional requirements

- Textual description
- System in its environment

Functional specification (MBD)

- Detailed description of dynamic behaviour and algorithms
- Traceability to functional requirements

Functional validation (MBT)

- Tests scenarios generated by MaTeLo are executed under QTE
Test Model – Time Management

- **Discrete time**
  
  - SSM::set Model/Input1 true;
  - SSM::set Model/Input2 false;
  - SSM::cycle 1;

- **Timed events**
  
  - set cycles [seconds * $tick]
  - SSM::wait $cycles
  
  - set output [SSM::get Model/Output]
  - checkOutput $output $expected
Test Model – Advanced Features

- Ignored inputs
  - Requirement: in state S (...), all other inputs are ignored
  - Exhaustive approach
    - Set “other” inputs one at a time
  - Stochastic approach
    - Draw random values for “other” inputs

- Random interrupts in a sequence of timed actions
  - Requirement: if signal S becomes active during the reloading sequence, perform steps 1 and 2, then continue sequence where it was interrupted
  - Stochastic approach ruled out (MaTeLo arcs coverage mandatory)
Tools Integration

- Test scripts are generated by MaTeLo in the TCL language
- MaTeLo uses SCADE QTE functions to set inputs, read outputs, handle time
- Requirement traceability is maintained
  - Requirements ID’s included in test scripts as comments
- Simple and powerful integration
  - Automated execution of tests
  - Seamless traceability of requirements

SSM::set Model/Input
SSM::get Model/Output
SSM::cycle or SSM::cycle $n

# Requirements: SEQD_CTL_08
MBT Preliminary Results

- Integrated execution chain

- Verification of requirements (Functional Validation)
  - Errors in the (fake) requirements used for this project were indeed found
  - Errors in both models were found as well

- 100% MC/DC coverage achieved for test model and design model
Conclusion

- Integration of Model-Based tools has significant added value
  - ... and is quite easy to achieve
- The combination of early prototyping with MBT makes it possible to verify requirements very early in the lifecycle
  - Inconsistencies and lacks in the requirements were easily found
- The same test scripts can (and will) be reused later to validate successive implementation refinements of this function
  - Thanks to a very elaborate integration environment
- The Connexion project goes on...