SysML based Design for Variability enabling the Reusability of Legacy Systems towards the support of Diverse Standard Compliant Implementations or Standard Updates: The Case of IEEE-802.15.6 Standard for e-Health Applications

SIMUTOOLS, August 2015
S. Meacham, F. Gioulekas, K. Phalp
Presentation Outline

- Problem definition
- Proposed solution
- SysML methodology and metamodelf for:
  - Package level
  - Block level
- Conclusions
- Future work
Systems and Simulations

Bridging the system level descriptions with detailed simulations

Systems abstraction for reusability

Systems abstraction for evolution and conformance to changing standards
Problem definition
Standards compliance

- Very important for a range of industries (e.g. Telecoms)
- Particularly for Wireless Area Networks

- Standard compliance for product interoperability and interconnectivity
Standards Evolution and Legacy systems

- Standards evolve
- Industries need to create standard compliant products with the less possible cost involved
- Need to reuse legacy systems and provide a mechanism to facilitate their evolution to new-standards compliant systems
Existing Solutions to this problem

- Existing work (evolving standard compliance) for software engineering

- For hardware systems, flexibility is more of a problem thus the problem more emerging and difficult to address
Proposed Solution
Model-based Systems Engineering (MBSE)

- With MBSE, we design the system at the model (system) level and produce code automatically from the models.

- MBSE raises the level of abstraction enabling easy reusability of legacy systems and extensibility at the model level.
Variability Modelling in MBSE

- Variability modelling is a technique that is usually used with MBSE and for industry sectors such as *car control systems*.

- Main idea is the design of *core* and *variants* in order to enable reusability and evolution of the system.

- Product line engineering
Methodology package variability

Methodology for modeling variability at package level

Case A (mandatory). We use «import» to include the core package to the variant package.

Case B (optional). We use «import» to optionally include the core package to the variant package.

Case C. We use inheritance to model alternatives with a “common inherited” functionality.
Meta-model for package variability
Methodology for block variability

Methodology for modeling variability at block level using SysML

Case A. We use composition to model that a block should be part of other blocks (core block)

Case B. We use aggregation to model that a block could be part of other blocks (optional block)

Case C. We use inheritance to model alternatives with a “common inherited” functionality
Meta-model for block variability

Metamodel for Block Diagrams with Variability

- **`diagram`**: Block Diagram for Variability
- **`graphic node`**: Core Block
- **`variant`**: Variant Block
- **`graphic path`**: Relationships
  - Generalisation (alternatives)
  - Composition (mandatory)
  - Aggregation (optional)
IEEE 802.15.6 Physical Layer classification
System with core and variant packages

System with core and variant packages

Core Package

<<import>>

Narrowband

<<import>>

Ultra wideband

<<import>>

Human Body Communications
Variants: Single pulse shaper & (ON-OFF Keying) Mapper of R=1/2
(R=1 for PHR) K=1 & M=2
Comparing with other approaches

- SysML is a standard in systems engineering so comparing to other formalisms it has wider applicability.

- In this paper, we are suggesting the use of a subset of SysML for simplicity and wider applicability.

- However, extending SysML will be necessary by our approach as well in order to produce better results (reusability and effectiveness of the proposed method).
Conclusions

- The problem of evolving standard-compliant systems can be addressed using..

- .. models to raise the level of abstraction to handle complexity and change with

- ..variability modelling to facilitate easy evolution

- SysML choice as the preferred standard option for systems engineering
Systems and Simulations

Bridging the system level descriptions with detailed simulations:

For performance modeling (Matlab models)

For requirements modeling and conformance to changing requirements (Matlab simulations)
Future work

- Connect SysML with Matlab simulations

- Thus, creating “simulatable” system description

- With the vision to create “evolving” or even “clever” models