Model-Based Software Engineering and Certification: Some Open Issues

Stefano Russo, Fabio Scippacercola
DIETI, Università degli Studi di Napoli Federico II
{stefano.russo, fabio.scippacercola}@unina.it

www.dessert.unina.it
www.critiware.com
Model-Based Engineering

- Critical systems demands for strong development efforts, and intense activities of verification and validation (V&V)
- Model-Based Engineering (MBE) and Model-Driven Engineering (MDE) support the development of modern complex systems, increasing the level of abstraction, and enabling the automation of development and V&V activities
- MDE is founded on models and transformations: the engineers build models, and artifacts are generated via transformations
Model-Based Engineering

- MBE proved to be effective in several application sectors, e.g. embedded systems
  - Most increments in software productivity are obtained increasing the abstraction level
  - We now have MBE languages, processes, standards, tools…
- MBE is appealing, but despite the “rather long” history …
- … insufficient empirical evidence of success in delivering its claimed benefits
- Not easy to find documented success stories
- “There are reports of improvements in software quality and of both productivity gains and losses, but mainly from small-scale studies” [Mohagheghi 2008]

Is it enough?

- “Any scientific approach to customize UML as a modeling notation for the use with safety-critical systems has not only to fulfil the intentions but also the practical certification-oriented requirements set by the standards” [Huhn 2010].

- Many UML profiles, MBE techniques, etc. in the literature (MBE jungle), but still at the stage of research efforts – need maturation and consolidation

- Tools have to be qualified
  - Currently, few tools are

- Standards are evolving
  - E.g., OMG SACM - Structured Assurance Case Metamodel (combines ARM and SAEM), July 2015

Some studies on State-of-the-Practice

♦ A survey in the Brazilian embedded software industry [Agner 2013]
  - 45% of the 209 participants use UML
  - increases in productivity and in quality, maintenance and portability
  - models mainly used for documentation
  - little use of code generation or model-centric techniques
  - MDE hindered by lack of expert UML knowledge

♦ A qualitative study in the car industry [Kirstan 2010]
  - earlier detection of errors
  - higher degree of automation and cost savings during the initial phases of development
  - interoperability between tools difficult

Generally: (a) too little evidence in order to generalize results; and (b) many different approaches used in industry


[Kirstan 2010] Evaluating costs and benefits of model-based development of embedded software systems in the car industry – results of a qualitative case study, ECMFA, 2010
Model-Based Engineering

- Recent surveys show that industries use MBE pursuing a variety of goals and that MBE is usually only partially applied
  - Improve quality, improve reusability, reduce development time, …
  - Mainly used for parts or components, rarely for entire systems
- Still not clear to what extent they are actually adopted in industry and whether they achieve the claimed advantages

- Notwithstanding this, MBE techniques are increasingly advocated for use in critical systems engineering
  - E.g., ISO 26262 states MB Design benefits and provides detailed guidance
MBE and Certification

- Certification of critical systems demands for providing clear evidences that requirements have been properly considered and addressed.
- Considering certification aspects, it is not clear to what extent model-based approaches are really adopted in industries, and if they actually provide the claimed benefits in quality and productivity.
- We discuss the relation between MBE and certification, and identify some open issues.
Domain Standards and Life-Cycle

- Safety standards (e.g., EN 50128 or DO-178C) do not define a specific process for the development, but request a well-defined methodology, with specific activities, practices and outputs.

- The process must take into account safety requirements throughout the whole lifecycle, and *evidence* is required, to show that they have been properly addressed *since initial planning*.

- According to the severity of a failure, the system functions are associated to subset of activities, techniques, roles, objectives and constraints that are mandatory, recommended, or not recommended to follow in the process.

- Higher levels of integrity demand more rigorous procedures and stronger assessments.
Domain Standards and Modeling (1/3)

- **Modeling** is explicitly recommended by some standards, but these require guarantees on the soundness of the languages, especially when semi-formal languages are used.
- Models can be used for several activities/goals:
  - Requirement specification, traceability, architecture, behavior …
- Arguments for a specific modeling language have to be provided to show its adequacy for the quality characteristics required for a specific artifact:
  - Formal semantics is often an issue
  - Brings back to the “old” discussion on formal methods in industry
The role of models in the process is “increasing” in recent standards:

- **DO-178C (2011)** in the *avionics domain* refers model-based development and verification to the supplemental document DO-331: models can be used for specification and design, and model simulation is allowed for some verification activities.

- **EN 50128:2011** in the *railway domain* includes modeling among the major techniques for requirements specification, architecture, and design.

- **ISO 26262** in the *automotive domain* states benefits of model-based design and provides detailed guidance.
Recent Standards and Modeling (3/3)

- It is up to the engineers to understand how to define model-based processes that are adherent to the standard, e.g. by following sets of rules or contemplating some specific requirements of the standards.

- This problem is relevant since most MBE approaches neglect the particular requirements of certifiable systems, and this limits their application in safety-critical domains.
Tool qualification

- Another concern is tool qualification, since standards require supporting tools be verified to be correct.
- Standards typically consider multiple levels of qualification, according to the effects on the final product of faults in the tool:
  - Tools may introduce or fail to detect errors.
- The costs for tools qualification might overweigh the benefits of MBE.
MBE tool support for critical systems

- Model-based software engineering tools are on the market since many years
  - Offering also certification kits for multiple standards
- However, to effectively use them in the light of certification, they should be synergic with mature, solid and standard-compliant (often legacy) processes
- Mature tools adopt non standard notations, and vendors suggest specific methodologies to use with the tool
  - Vendors claim to have complete solutions, but …
  - Companies are not keen to change their legacy processes, and wish to save their past investments
- The costs of licenses for proprietary certified/certifiable tools and the lack of open and stable tool chains are still seen by many companies as obstacles to adopt MBE
MBE and provision of evidence

- MBE can provide effective support to produce safety cases:
  - Models can specify and trace requirement, while automatic transformations can generate reports, information, and artifacts, which structure the evidence
  - MDE approaches can assess if the artifacts fulfill the compliance requirements of the applicable safety standards
  - Models can also contribute to the safety evidence when risks have been properly considered and addressed by model-driven approaches (e.g. model-driven FMEA)
Most MBE *research* underestimates issues deriving from domain-specific certification standards, and from the integration into legacy industrial processes.

It is difficult to understand applicability, costs and benefits of MBE techniques for systems to be certified.

There are still too few *empirical studies* in the area, that help to understand the risks, as well as the impact that MBE approaches have, also on people and organizational factors.

- All this reduces the practical usage in industry
- Many companies still consider MBE risky

Very few studies focus on the impact on current industrial practices where product certification is pursued.

*We lack a real corpus of success and failure stories*
Open Issue (2/3)

- MBE is often considered by practitioners as the push-button solution, but *MBE has to be carefully integrated in consolidated processes to effectively support certification*

- Modeling languages should be adapted to the needs of critical domains (e.g., by domain-specific UML profiles)
  - Proprietary notations (e.g. SCADE, Mathworks) effective but non-standard
  - Standard notations (e.g. UML) see a proliferation of dialects, thus of supporting tools (often short-lived)
Open Issue (3/3)

- The availability, maturity and openness of tools play a crucial role for the adoption of model-driven technologies
  - Companies want to avoid vendor lock-in problems
  - Support tools (and companies) need to be long lived
- Matlab is nowadays the most used modeling environment for embedded systems, but recent focus is on standard modeling languages and open tools (e.g., UML and MDA)
  - The success of tools like Simulink and SCADE for safety-critical systems deserves more investigation, since it can provide useful insights for the improvement of UML-based techniques
- Finally, since software contributes to safety in a system context, we argue that SysML should take a prominent role for the MDE of certified systems
Conclusions

- Certification poses further challenges onto MB sw engineering
- MBE can provide real benefits towards software certification, as it supports reasoning at an increased level of abstraction, and it offers powerful forms of automation
- Best if MBE used both in software development and V&V
- However …
- More success (and also failure) stories and empirical studies are needed to show how and to what extent this can be achieved.
- If MBE methodologies and tools are still sometimes claimed to be a sort of *silver bullet* for software engineering, they are not (yet) the silver bullet for software-intensive systems certification