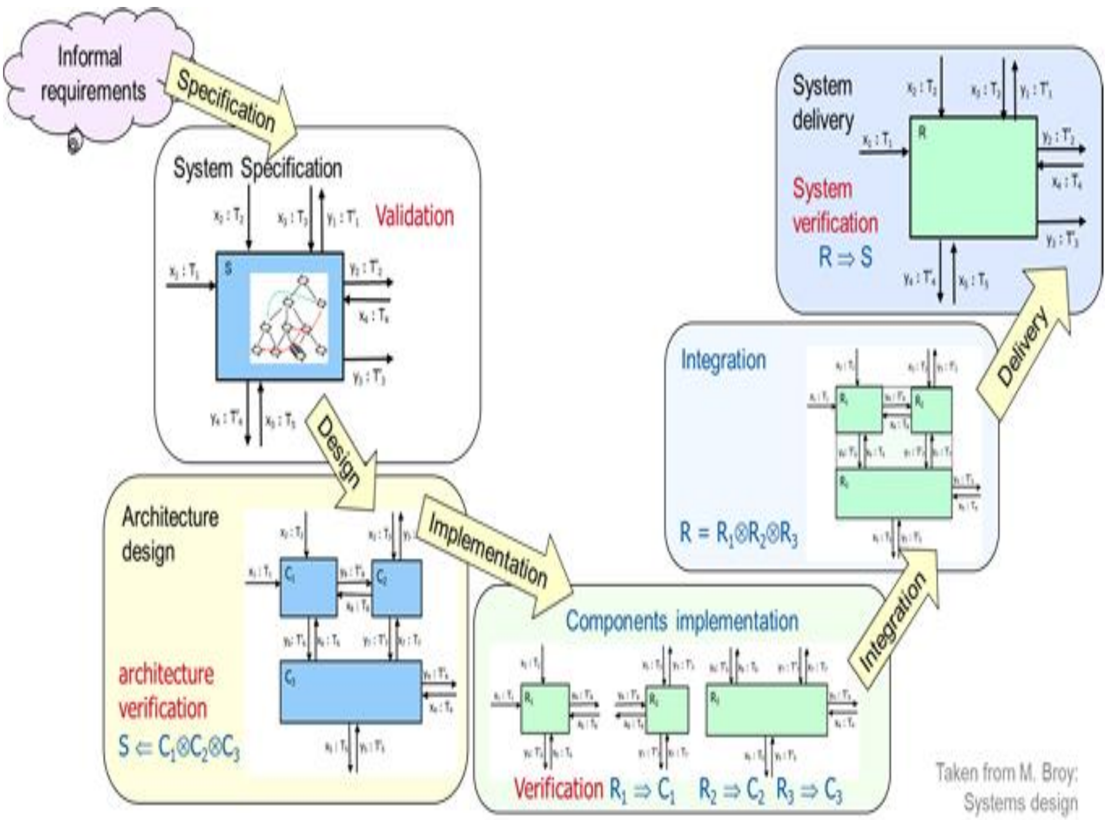
Leveraging AI Methods for Reducing Certification Costs of Safety Critical Cyber Physical Systems

Hendrik Kausch 1[1)], Mathias Pfeiffer 2[1)], Deni Raco 3[1)], Bernhard Rumpe 4[1)]
[1)]RWTH Aachen University

Avionics is definitely a safety-critical application domain. Software complexity is ever increasing together with more autonomy as well as increased real-time based interaction between airplanes, drones and potentially future air taxis. This again raises the question, whether developing software the same way as we did the last 30 years is still appropriate, or in the times of much better formal methods and cheap and powerful computational capabilities, it would be feasible to use clear and model-based specification techniques for an integrated systems engineering approach and formally verify any physical and logical implementation of functionality, including the software against that specification. This could be another important step towards quicker development of highly safety critical systems

Model-based approaches [3, 4] have had promising results in the recent years in managing the increasing complexity of system development. However, a modular design methodology is required to achieve scalability by compositional verification, by breaking down the complexity of verification activities. Furthermore, the transfer of successful AI-based approaches from KR&R (Knowledge Representation and Reasoning) can significantly reduce certification costs of critical systems by accompanying refinement steps with (generated) correctness-proofs.



This work presents such an approach being successfully applied [6, 7] for achieving a modular development of a safety critical cyber physical system. The treated case study of an avionics protocol was adapted from a NASA report on a collaboration of NASA and Rockwell Collins.

The modeling language used to design the avionics protocol is the Object Management Group Systems Modeling Language [8]. OMG SysML is a general-purpose graphical modeling language for specifying, analyzing, designing, and verifying complex systems that may include hardware, software, information, personnel, procedures, and facilities. In particular, the language provides graphical and textual representations with a semantic foundation for modeling system requirements, behavior, structure, and parametrics, which is used to integrate with other engineering analysis models.

In this case study a Pilot Flying System is modularly decomposed, its components are refined separately until an implementation is reached, and after composing back, one can be certain that the system behaves correctly by construction. The approach uses the dataflow-based mathematical methodology FOCUS [2, 3] to give semantics (in the sense of David Harel and Bernhard Rumpe in [1]) to SysML models. Underspecification is represented by sets of stream processing functions and refinement the simple concept of the subset operator on sets of stream processing functions. FOCUS stands out among competitors due to the fact that refinement is compositional.

Reasoning over critical systems is performed by mapping SysML in an encoding of a dataflow logical knowledge base in the theorem prover Isabelle [6, 7]. Proofs of correct refinement steps (or counterexamples for wrong ones) are generated. The verification of a property is thus reduced to an intelligent (proof-) search problem. This approach can very well be used also in cybersecurity as a preventive vulnerability detection at design time.

[1] D. Harel, **B. Rumpe**: *Meaningful Modeling: What's the Semantics of "Semantics"?*, In: **IEEE Computer**, Volume 37, No. 10, pp 64-72, IEEE, October 2004, https://www.se-rwth.de/staff/rumpe/publications20042008/Meaningful-Modeling-Whats-the-Semantics-of-Semantics.pdf

[2] J. Ringert, **B. Rumpe**: *A Little Synopsis on Streams, Stream Processing Functions, and State-Based Stream Processing*. In: **International Journal of Software and Informatics**, Volume 5, 2011, https://www.se-rwth.de/publications/A-Little-Synopsis-on-Streams-Stream-Processing-Functions-and-State-Based-Stream-Processing.pdf

[3] W. Böhm, M. Broy, C. Klein, K. Pohl, **B. Rumpe**, S. Schröck (Eds.):. *Model-Based Engineering of Collaborative Embedded Systems*, **ISBN 978-3-030-62135-3**. Springer, Jan. 2021, https://www.springer.com/gp/book/9783030621353

[4] **B. Rumpe**. *Modeling with UML: Language, Concepts, Methods*, Springer International, **ISBN 978-3-319-33933-7**, July 2016, https://link.springer.com/book/10.1007/978-3-319-33933-7

[5] A. Butting, O. Kautz, **B. Rumpe**, A. Wortmann: *Continuously Analyzing Finite, Message-Driven, Time-Synchronous Component & Connector Systems During Architecture Evolution*. In: **Journal of Systems and Software**, 2018, https://www.se-rwth.de/publications/Continuously-Analyzing-Finite-Message-Driven-Time-Synchronous-Component-and-Connector-Systems-During-Architecture-Evolution.pdf

[6] **H. Kausch, M. Pfeiffer, D. Raco**, **B. Rumpe**. *Model-Based Design of Correct Safety-Critical Systems using Dataflow Languages on the Example of SysML Architecture and Behavior Diagrams*, Proceedings of the Software Engineering 2021 Satellite Events, Braunschweig/Virtual, http://ceur-ws.org/Vol-2814/paper-A4-5.pdf

[7] **H. Kausch, M. Pfeiffer, D. Raco**, **B. Rumpe**. *MontiBelle - Toolbox for a Model-Based Development and Verification of Distributed Critical Systems for Compliance with Functional Safety*, **American Institute of Aeronautics and Astronautics**, Modeling and Simulation Technologies, 2020

[8] The Object Management Group Systems Modeling Language SysML, https://www.omgsysml.org