

# Generation of Test Strategies for Model-based Functional Safety Testing using an Artifact-Centric Approach

Bert Van Acker, Joachim Denil, Paul De Meulenaere  
CoSys-lab

University of Antwerp, Belgium  
Flanders Make, Belgium

Bert.VanAcker@uantwerp.be, Joachim.Denil@uantwerp.be, Paul.Demeulenaere@uantwerp.be

Bjorn Aelvoet, Dries Mahieu, Jan van den Oudenhoven

DANA BELGIUM NV

bjorn.aelvoet@dana.com, dries.mahieu@dana.com, jan.vandenoudenhoven@dana.com

## Abstract

**ISO/DIS 26262, an automotive functional safety standard, provides stringent requirements and processes for a 'safety-oriented' software lifecycle and in particular on the verification and validation part. These strict and activity-based safety processes impose some important drawbacks, especially with the increasing complexity of software intensive safety critical systems. In this paper we report on a methodology for guiding the Model-based Functional Safety testing by generating valid test strategy models. We explicitly model test artifacts and process rules, which allows to automatically generate valid and optimized test strategies for Model-based Functional Safety testing. A well-known advanced driver assistance system, the adaptive cruise control, is used to demonstrate the proposed methodology.**

## 1 Introduction

Development of software intensive safety critical systems, such as found for example in the avionics, automotive, maritime, and energy domains, is becoming increasingly complex. This increasing complexity is not only caused by the synergistically interaction

between software and physical elements[Lee08], it is also caused by the safety aspects of such systems. Safety engineering is a cross-cutting concern that is taken into account throughout the complete life-cycle. Safety engineering aims to show that the required system functionality is safe and reliable[Bellotti10]. This way, the quality and reliability of safety critical systems are highly dependent on both the implementation and proper system validation and verification. To cope with this complexity, engineers can adapt model-based engineering principles, enabling complex system analysis via system simulations. Simulations allow for a preliminary validation in very early stages of the development process, this reduces the risk of high redesign cost by detecting premature errors. Moreover, safety critical systems must adhere to functional safety norms like IEC 61508[IEC61508] and ISO/DIS 26262[ISO/DIS26262]. These standards pose stringent requirements for development of safety critical systems and also on the testing processes. Adhering to these standards can be difficult because the defined activity-centric processes are rigid and some margin for appreciation can exist, so implementing an instantiation of the requirements outlined in ISO/DIS 26262 within a Model-Based Design requires special consideration[Conrad12]. Moreover, the verification and validation (V&V) processes, which contain not only testing activities but also e.g. (static) analysis, inspections and reviews, etc., are preferably application-specific meaning that the instantiation of the V&V process is fitted for the application in development, which can reduce the V&V overhead.

In our approach, we propose to focus on the artifacts instead of the fixed process flows outlined in the ISO/DIS 26262. Artifacts are the products *used by* and/or *generated from* process activities and the resulting artifact-centric process models are defined as acceptable states of the process without enforcing any specific execution flow[Baresi16]. Process rules are used to formally capture the constraints of a valid process execution flow, enabling us to (semi-)automatically generate valid, optimized and customized test strategy processes for Model-based Functional Safety testing. Process rules can originate from various sources such as company-specific process knowledge or safety-specific knowledge captured in the functional safety standard. By applying these process rules at different steps in the workflow, we can truly make the test strategies application- and company-specific, facilitating the usage of the test strategy processes and further reducing the V&V overhead.

The rest of the paper is organised as follows : Section 2 presents some essential background and related work on functional safety testing and the used artifact-centric approach. Section 3 discusses the basic concepts of the presented solution. Section 4 implements a case study and finally, Section 5 concludes the paper.

## 2 Background and related work

Safety is one of the key issues in future road vehicle development[Bellotti10] and despite the increasing complexity, caused by new complex driver assistance systems with e.g. accident prediction and avoidance capabilities and the associated complicated processes, car manufacturers have the basic obligation to only put safe products on the market. In safety critical systems, human safety depends upon the correct operation of those systems. This raises the need for development methods and processes that could lead to provable correct systems[Arc17]. Therefore, car makers as well as suppliers adhere to a functional safety standard, namely the ISO/DIS 26262. This standard gives stringent requirements for a 'safety-oriented' design and in particular on the verification and validation part. A high-level guidance to design proper safety functions is provided in the guidelines, together with processes to verify and validate these safety functions and prove their adherence to the standard. The aforementioned safety functions are functions implemented in a control system to prevent violation of the safety goals. These safety functions will assure safe operation of the system and will put the system in a safe state in case of an inevitable safety goal

violation. A clear example of a safety function is a *safe standstill function* in road vehicles which will monitor the vehicles movement during standstill and will activate the park brake if an unintended movement occurs.

This guidance is activity-based, meaning that a chain of activities is provided and the control flow is well defined and fixed. This works very good for defining the normal execution flow and the management of foreseen exceptions during execution of the processes but when unforeseen situations occur, the correct execution within the process definition cannot be checked[Baresi16]. Artifact-centric processes could provide a good solution to handle these unforeseen situations as it provides a process definition in terms of acceptable process states and not enforcing a specific execution flow. [Yongchareon18] states that artifact-centric process modeling has been evidenced with higher flexibility over traditional activity-centric process modeling and they used it to improve inter-organizational business cooperation. [Kuhrmann14] also states that by focusing on the artifacts, which precisely define the desired outcomes, rather than on specific methods, the processes are less generic and more fitted for the organization.

## 3 Approach

This section introduces the proposed methodology of our approach, which is shown conceptually in Figure 1.

The result of our approach is a (semi-)automatically generated valid and optimized test strategy process compliant to the ISO/DIS 26262 functional safety standard. We start from an integration model(top-left in Figure 1), where a set of components are connected together to form a particular safety function. Each of these components and their integration need to be tested compliant the functional safety standard. These components can originate from different sources such as an internal software department or external suppliers and can already been tested partially or completely.

This knowledge about the level of test completion is captured for each component <sup>1</sup> within the artifact-view of the component. The corresponding artifact-view of the components are shown in the top-middle of Figure 1 as parallelograms with four squares at the bottom. These squares depict the different integration levels of the system under test (SUT) within the model-based testing of embedded systems, namely Model-in-the-Loop (MiL), Software-

---

<sup>1</sup>With 'component', we intend to denote the components of the integrated system and the integrated system itself



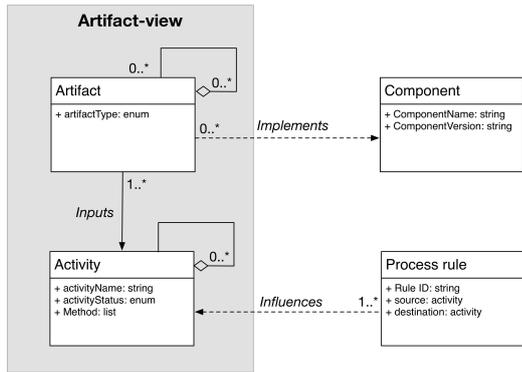


Figure 2: Artifact-view meta model and dependencies

- **Activity** class defines an activity. The status is a key element captured within this artifact-view.

In the scope of functional safety testing, where the possible V&V test methods are imposed by the ISO/DIS 26262 standard, the method attribute in the activity object enables the customization of the test strategy by selecting one or more available V&V test methods. The process rules on the other hand will have influence on the activities, e.g. they can alter the available V&V test methods or omit an activity depending on other activities. The process rules will also define the control flow between the activities, deemed necessary to generate a valid test strategy process. Note that an artifact-view can contain links to other artifact-views, indicating that the artifact-view corresponds to an integration model of different components.

In our approach, the artifact-view model is a text-based model defined with an extensive mark-up language (XML). This enables the model to be human- and machine readable, which is deemed necessary to process this model in the subsequent steps of the workflow.

### 3.2 Process rules

The key element of our proposed approach is the explicit modeling of process rules which enables the optimization and customization of the test strategy processes. These rules can be applied at different stages of the workflow, depending on the rules and their impact. As mentioned before, rules can e.g. originate from requirements posed by standards or company-specific decisions. Important benefit of explicitly modeling these process rules is that all process rules or decisions are formally captured and can be used as well-documented evidence to prove process compliance to certain standards.

Figure 3 is a graphical example of a process rule

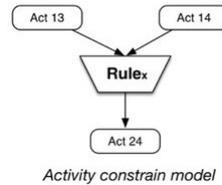


Figure 3: Graphical example of a process rule

where activity 13 and activity 14 induce a particular rule to activity 24.

Following rules can be applied:

- **Sequential execution** of activities
- **Parallel execution** of activities
- Activity **not feasible** within valid process
- **Preferred V&V method** for activity
- V&V method selection **forces V&V method** for other activity

Note that these set of rules can easily be extended with extra process rules. An example of a more advanced process rule, originating from both the safety standard and company-specific knowledge is defined as follows: if the functional behavior of a component is tested via a *requirements-based test method* on **model level (MiL)**, we force the V&V test method on **software level (SiL)**, namely a *back-to-back test method* combined with an *interface test method*.

Up till now, we used a text-based model to define the process rules. This could be improved by defining a graphical domain-specific language (DSL), which eases the modeling of these process rules.

### 3.3 Transformations

With the artifact-views and the process rules defined, we can generate a set of valid test strategy processes. This **transformation** extracts the meta-information of all artifact-views and generates a process model where the appropriate process rules are taken into account. The generated process model can either be a directed graph or a Causal Block Diagram representing a (executable) process model. This process model can further be customized by selecting the proper V&V test methods and again applying the specified process rules. This will be further explained by means of a practical example in Section 4.

## 4 Case Study

In this section, we use a safety critical adaptive cruise control system as academic use case to illustrate the proposed concepts of the previous section. The

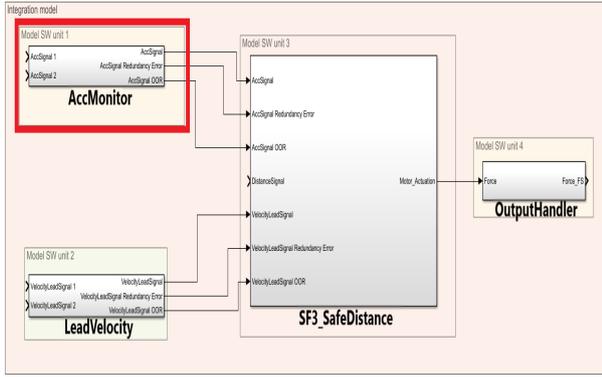


Figure 4: SafeDistance safety function

adaptive cruise control system is an advanced driver assistance system (ADAS) which is already widely available in commercial road vehicles but interesting as system under study because it is one of the precursors of fully autonomous vehicles[Nardi17]. For this use case, we will decrease the complexity by focusing on one safety function defined in the implementation of the adaptive cruise control, namely the SafeDistance function. The safety chain for this SafeDistance safety function is shown in Figure 4.

This safety chain is decomposed in four components, each on a different level of abstraction. More specifically, each component is unit tested up till a certain test integration level. This knowledge is formally captured in the corresponding artifact-view of each component. The artifact-view of the complete SafeDistance safety function is shown graphically in Figure 5. This defines that the AccMonitor component, leftmost artifact-view in Figure 5 and highlighted in the red box in Figure 4, is unit tested up till the software integration level. To complete the unit test for this component, the tests on the processor and hardware integration level need to be performed. From this graphical representation, the artifacts are not explicitly defined<sup>2</sup> as they are implicit present in the activity names, namely:

- **Model-in-the-Loop:** Input artifact is a *model* of the AccMonitor
- **Software-in-the-Loop:** Input artifact is *source code* of the AccMonitor
- **Processor/Hardware-in-the-Loop:** Input artifact is *production code* of the AccMonitor

Besides the necessary knowledge about the test completion, we also define general process rules for a valid execution flow of the model-based test activities. We define that (i)SiL needs to be sequentially

<sup>2</sup>Artifacts are also not explicitly present in the processes

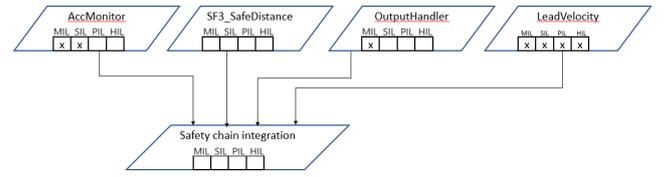


Figure 5: Safety chain schematic artifact-view

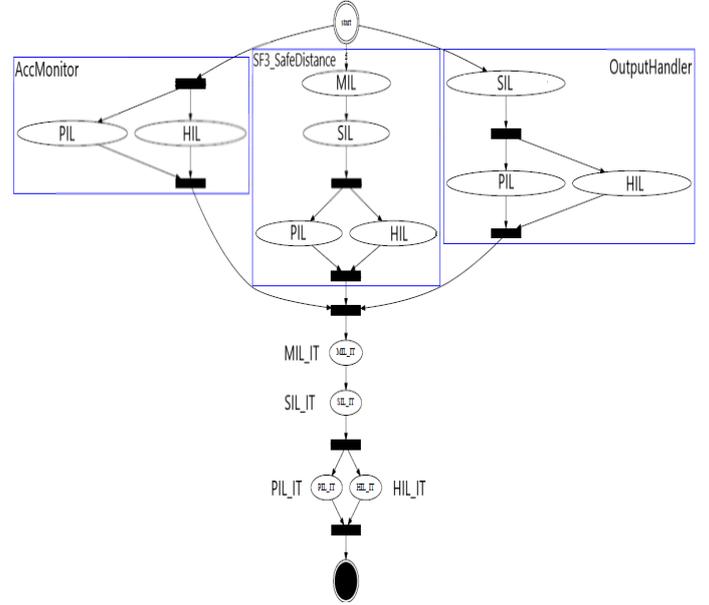


Figure 6: Model-based functional safety test process

performed after MiL, (ii)Pil and Hil need to be sequentially performed after SiL, (iii)Pil and Hil can be performed in parallel and (iiii) integration tests need to be sequentially performed after the completion of the unit tests of all integrated components.

With the artifact-views and the process rules defined, a valid test strategy process model can automatically be generated. This text-to-model transformation generates a directed graph, as shown in Figure 6, where the test activities are depicted as nodes and the control flow as links.

In this use case, we not only generate a directed graph, we also generate a process model in a Causal Block Diagram (CBD) formalism in Simulink©, which allows us to visually select the proper V&V test methods for the test activities, when necessary, to truly customize the test strategy process. An example of a test activity in CBD formalism is shown in Figure 7, where the selection of the V&V test methods, if applicable, is present for the sub-activities. The available V&V test methods and the possible test case derivation methods are compliant to the ISO/DIS 26262 as shown by the *SpecifiedBy* relation between the tables,

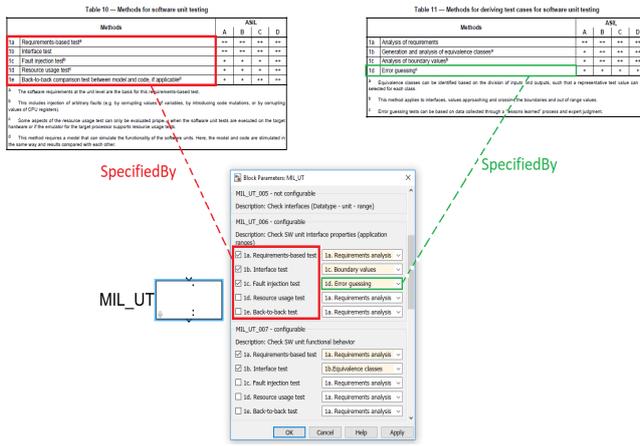


Figure 7: Configurability of functional safety test activity and ISO/DIS 26262 compliance relation

originating from ISO/DIS 26262 - part 6, and the customizable CBD block.

At this level, process rules are defined to capture the dependencies between the different test activities and their corresponding V&V test methods. The following list is a subset of the outcome of the applied process rules for this use case:

- Resource usage test **not feasible** on MiL
- Back-to-back test **not feasible** on MiL
- Requirements based test on MiL **forces** requirements based test and boundary values test on SiL
- PiL **not feasible** for unit testing
- HiL **not feasible** for unit testing

After selecting the proper V&V test methods<sup>3</sup> and applying these process rules, a valid, optimized and customized test strategy model is automatically generated.

An example of a generated test strategy model is shown in Figure 8. In this generated test strategy process, the proper V&V test methods are selected and the above mentioned process rules are applied. To increase the readability and usability of the generated test strategy model, we grouped the sub-activities of each test activity at the particular integration level.

<sup>3</sup>The applied process rules have also influence on the available/valid V&V test methods

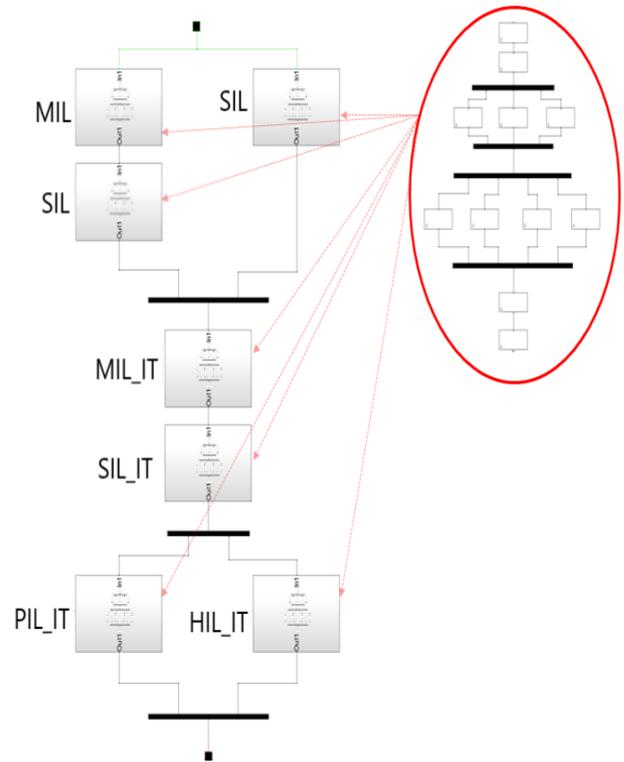


Figure 8: Generated valid, optimized and customized test strategy

## 5 Conclusion and future work

This paper presents a methodology to facilitate the model-based validation and verification of safety critical systems by (semi-)automatically generating valid, optimized test strategy processes compliant to the ISO/DIS 26262 functional safety standard using an artifact-centric approach. We applied this methodology to a well-known advanced driver assistance system, the adaptive cruise control, extended with the needed safety functions. This case study is small-scale and yet complex enough to be suited for future research.

In the future, we plan to extend the proposed methodology by introducing one or more design space exploration (DSE) algorithms to further optimize the generation of valid test strategy processes. By taking the resource constraints, such as shared real-time hardware platforms or human resources, into account, both a valid and optimized test strategy process and an optimized test scheduling can be generated, which is beneficially for the overall verification and validation processes.

Second, we want to extend the usage of the artifact-view by tightly coupling this artifact-view to the model information or meta-knowledge [Sirin14],

enabling the use in design processes other than safety-related automotive development. More specifically, we will include this information of the artifact-view within the validity frame [Denil17][Klikovits17] encapsulated with each component. This will increase the (re-)usability of the components within the design processes.

Lastly, we will compare our proposed methodology against more traditional activity-based processes to empirically assess the usability, correctness and scalability of our methodology.

## 6 Acknowledgments

This work has been carried out within the framework of Flanders Make's ICON project 'EnPower' (TCO optimal system design for Energy and Power storage in dynamic load applications, HBC.2016.0462) funded by the agency Flanders Innovation & Entrepreneurship (VLAIO) and Flanders Make. Flanders Make is the Flemish strategic research centre for the manufacturing industry.

## References

- [Arc17] Arcaini, P., Gargantini, A., & Riccobene, E. Rigorous development process of a safety-critical system: from ASM models to Java code. *International Journal on Software Tools for Technology Transfer*, 19(2), 247269. <https://doi.org/10.1007/s10009-015-0394-x>, 2017.
- [Baresi16] Baresi, L., Meroni, G., & Plebani, P. On Handling Business Process Anomalies through Artifact-based Modeling, 2016.
- [Bellotti10] Bellotti, M., & Mariani, R. How future automotive functional safety requirements will impact microprocessors design. *Microelectronics Reliability*, 50(911), 13201326. <https://doi.org/10.1016/j.microrel.2010.07.041>, 2010.
- [Conrad12] Conrad, M. Verification and Validation According to ISO 26262 : A Workflow to Facilitate the Development of High-Integrity Software. *roc. ERTS 2012 Embedded Real Time Software Ans Systems.*, 2012.
- [Denil17] J. Denil, S. Klikovits, P. J. Mosterman, A. Vallecillo, and H. Vangheluwe, The experiment model and validity frame in M&S, *Proceedings of the Symposium on Theory of Modelling and Simulation*, 2017.
- [IEC61508] IEC 61508 ed. 2.0. International Electrotechnical Commission, *IEC*; 2009.
- [ISO/DIS26262] ISO/DIS 26262 (all parts), road vehicles functional safety. International Organization for Standardization.
- [Klikovits17] Klikovits, S., Denil, J., Muzy, A., & Salay, R. Modeling frames *MoDeVVa Workshop Proceedings*, 2017.
- [Kuhrmann14] Kuhrmann, M., & Beecham, S. Artifact-Based Software Process Improvement and Management : A Method Proposal. 2018.
- [Lee08] E. Lee et al., Cyber physical systems: Design challenges, *Object Oriented Real-Time Distributed Computing (ISORC), 2008 11th IEEE International Symposium on. IEEE*, pp. 363369, 2008.
- [Nardi17] Nardi, A., & Armato, A. Functional safety methodologies for automotive applications. *IEEE/ACM International Conference on Computer-Aided Design, Digest of Technical Papers, ICCAD, 2017Novem, 970975*. <https://doi.org/10.1109/ICCAD.2017.8203886>, 2017.
- [Sirin14] G. Sirin, C. J. Paredis, B. Yannou, E. Coatanea, and E. Landel, A Model Identity Card to Support Simulation Model Development Process in a Collaborative Multidisciplinary Design Environment, 2014.
- [Yongchareon18] Yongchareon, S., Jian, Y., & Zhao, X. A View Framework for Modeling and Change Validation of Artifact-Centric Inter-Organizational Business Processes, *Information Systems (April)*. <https://doi.org/10.1016/j.is.2014.07.004>, 2018.
- [Zander11] Zander, J., Schieferdecker, I., & Mosterman, P. A Taxonomy of Model-Based Testing for Embedded Systems from Multiple Industry Domains. *Model-Based Testing for Embedded Systems*, 122. <https://doi.org/doi:10.1201/b11321-2>, 2011.