

# **The “Standards” Way of Modeling**

## **How I Learned to Stop Worrying and Love Modeling**

Robert Baillargeon, Sodus  
Walter van der Heiden, Willert Software Tools

**With every new standard there is a target of an improved process of engineering. However, to design vehicles no standard is used, or can be judged, in isolation. For this reason, it is critical that we have patterns of adoption that leverage these standards to their strength in concert with other standards. Over decades of practice in engineering design and tool deployment our teams have identified patterns of delivering the best from standards that improves adoption for the engineers.**

### **Challenge with Standards**

The task of engineering, and engineering vehicles, is filled with standards. These standards are targeted to communicate best practices, improve quality, and enhance the overall engineering experience for the better of the industry. In almost all cases these objectives are indeed reached. However, at the individual engineer level it is not always clear what standards apply and how best to leverage them to achieve the task at hand. As automotive engineers we are often required to become knowledgeable at multiple standards and individually weave them together to plan our engineering tasks.

Our team has found that the individual interpretation of standards has led to confusion and the inverse effect of maximizing the engineering efficiency. We have found that the leveraging of these standards can be navigated providing value and freedom to engineer vehicles more effectively and with better results. Our method is focused on the imperatives of the standards, focus on the flow of information, and focus on the tasks of the Engineer.

### **Our Objective for Flow**

While each of our standards is beneficial, and likely necessary, to our vehicle engineering the challenge is how we leverage the standards to enhance the flow of information between the engineering activities. Our objective in flow is twofold; avoid the repetition/recreation of assets and naturally build the connectivity/traceability in assets that makes our design process robust. Together this flow builds a thread of engineering connecting and tracing the assets of design.

Our motivation in the usage of standards is to leverage their best practices. Our challenge is these best practices are often focused on a narrow set of challenges that do not fully encompass all of the challenges of vehicle engineering. Indeed, as tool integrators we observe these same challenges in bringing tools together.

To solve these problems, both in theory and practice, we focus on what we believe are the imperatives to flow. First, the intersection between tools and standards and how they present multiple overlapping solutions to the problem space. Second, the transition between tools and data formats which inhibits how we may flow

information between tools and activities. Our goal is to make the transition a natural flow that amplifies the best in both standards and tools.

### **Organizing Standards**

In our method of flow, we identify the importance of standards but further characterize standards for the role they play. Effectively there are three categories of standards we organize by:

- Standards of Process & Methods – These are standards that define the engineering practice, but not the form of execution.
- Standards of Interchange & Connectivity – These are standards that define how we may connect information and engage tooling.
- Domains Standards – These are standards define an engineering practice or activity on development of and engineering asset.

This layering approach of standards enables us to identify the engineering process flow holistically and integrate the domain specific assets in a synchronized configuration of standards.

### **Common Standards in Automotive**

When interacting in the Automotive community a multitude of standards can be applied. To focus our discussion with examples we are utilizing the following standards in specific roles.

- The process of engineering is driven by the practices of *Automotive SPICE*. In this role, ASPICE outlines the needs of the engineering artifacts and the roles these artifacts play in the development of our assets.
- The deliverables of engineering are guided by the *AUTOSAR* standard and its exchange format of ARXML. In this role AUTOSAR outlines the design language of a vehicle architecture and a node architecture.
- The practice of Systems Engineering is outlined by *SysML*. The expressions of SysML provide the palette of refining practices to move from use cases to design blocks.
- The practice of Software Engineering can be exhibited in the architecture designs expressed in *UML*. As a natural language of the Software Engineer, we can observe the transition from software concept to implementation.
- The practice of Tool Integration can be formulated by web standards and more specifically Open Services for Lifecycle Collaboration. The *OSLC* standard provides a method to discover and link a diverse set of tools in consistent relationships that support engineering.

### **Application in Practice**

To understand the pattern of integration suggested, we have assembled a pragmatic application of standards in the Automotive context.

Automotive SPICE is the foundation of process practice in Automotive today. While not an executable process, it identifies the practices and the deliverables expected from sound automotive engineering practices. From this standard, we identify the

core systems and software engineering practices along with supporting processes such as change and configuration management. We perceive a critical component of ASPICE as the mandated traceability between assets. While ASPICE does not specify the method of linking assets, the OSLC standard is targeted at connecting repositories and providing linkages between artifacts. By mandating our repositories utilize OSLC (or an OSLC proxy of elements), the foundation of our assets can be linked and navigated from a source thru a set of targets and achieve the intent of ASPICE to enable impact analysis on any change.

To satisfy the Systems Engineering objectives of ASPICE we utilize SysML to define the application content. Using a standards-based flow enables the capture of use cases, interactions, interfaces and blocks in an evolutionary process and also enables growth of understanding and depth of design that refines the target behavior. The challenge with adoption of SysML has been the inability to transition to the language of design in Automotive. To address this challenge, we leverage a SysML to AUTOSAR bridge which translates SysML block elements to AUTOSAR Software components and seeds and continues the design activity while providing traceability.

In continuing our mission of traceability and accessibility, we project AUTOSAR data via a publisher to an OSLC Repository. While ARXML is useful for the exchange of information, it does not provide the linkable foundation that is valuable to support linking traceability. With the projection of the AUTOSAR design, the assets and relationships native to AUTOSAR become available for traceability. As the AUTOSAR design evolves, including allocation and network configuration, this information is now traceable and navigable enhancing the value of the investment.

To satisfy the Software Engineering objectives of ASPICE we need to capture the Architecture and Design of our Software. AUTOSAR simplifies part of this task by mandating a standard run-time architecture. The software architecture design can now be the software component designs, runnables, and interfaces. UML provides a natural software language to express the design that is traceable and linked when published to an OSLC repository. Additionally, Using UML in conjunction with code generation fulfills the full lifecycle from our initial requirements/SysML model down to the lowest level design elements.

Reflecting our value of layered approach to integrating standards we use the process, ASPICE, as the foundation of describing the objective of the engineering activities. In the middle, the OSLC standard provides the framework for us to define and resolve the interaction between standards-based descriptions. And finally, we use AUTOSAR, UML, and SysML as domain specialization standards that can leverage OSLC to meet their roles in the ASPICE context for traceability.

## **Conclusion**

The benefit of this design philosophy is that we are able to use dominant standards and standards-based tools for the engineering tasks presented to us. Leveraging OSLC and the ability to build and maintain links thru a diverse set of repositories allows us to build a thread of engineering that meets the expectations of ASPICE and exceeds the usability and maintainability of traditional methods. Additionally, thru the usage of standards-based integration we can leverage the value of modeling and

continue the flow not just in the modeling tool, but the entire development process. Our innovation and novelty is on the integration and intersection of standards to enable more effective engineering.

### **References**

The Open Services for Lifecycle Collaboration. <https://open-services.net/>

Automotive SPICE. <http://www.automotivespice.com>

AUTOSAR. <http://www.autosar.org>

SysML and UML. <http://www.omg.org>

### **Authors**

**Robert Baillargeon.** Senior Systems Developer with wide experience in Automotive Systems and Software Design. He is Vice President of Engineering at Sodius, a leading company in the development of integration tools for engineering.



### **Contact**

Email: [rbailargeon@sodius.com](mailto:rbailargeon@sodius.com)

**Walter van der Heiden.** Senior Software Developer with a wide range of experience in the embedded software field. Since 2001 CTO/ Senior Partner of Willert Software Tools, a leading company in the area of embedded development tools.



### **Contact**

Email: [wvdheiden@willert.de](mailto:wvdheiden@willert.de)