The Best of Both Worlds

Product Structure and Functional Representation

Josef Zehetner, Katrin Moser, AVL

The automotive industry is currently facing two overarching challenges: increasing complexity and the need for speed concerning time-to-market. In addition, we see a stronger shift towards software and a decoupling of the development of hardware and software. The complexity of software will increase as vehicle functions and features are determined more and more by software. The vision needed to address these challenges must focus on leveraging existing data and expertise, increasing efficiency and enabling more agile development overall. This applies not only to OEMs and automotive suppliers, but can also be extended to many other industries and product-oriented companies. One question still remains: How do we establish a continuous exchange between product-structure-driven systems (e.g. BOM, PDM, etc.) and a functional representation (Figure 1).

Physical prototypes are quite expensive, they are available only very late in the development process and they are a scarce resource. Besides, various departments have to rely on simulation and testing to perform their development, verification or validation but the resulting information is often shared only informally via emails or other documents. This leads to a lack of clarity, consistency and traceability of data as well as long response times. What's more, each department wants to use the most suitable tool for its development task(s), resulting in a heterogeneous tool and system landscape all in all.

The challenge of interoperability is neither trivial nor can it be easily solved with a single tool. So how can we enable the consistent sharing of gathered data and re-use of the relevant information objects, e.g. models, parameters, result data or expertise? Real-life implementation requires the integration of fundamental principles, neutral connectors, connecting solutions and will also generate organizational changes, e.g. new roles or departments for continuous verification and validation (V&V). This might be reflected in a centralized methodology department with model managers, parameter managers or virtual prototype responsibles. The goal is a homogeneous workflow where virtual and physical components are seamlessly interchangeable.

In this context, we apply a set of principles to bring to life our vision of continuous V&V for a product under development: requirements engineering and management, systems engineering, model-based systems engineering and product-lifecycle management. Additionally, an Integrated and Open Development Platform (IODP) brings simulation and test together and enables continuous validation. Many interoperability problems

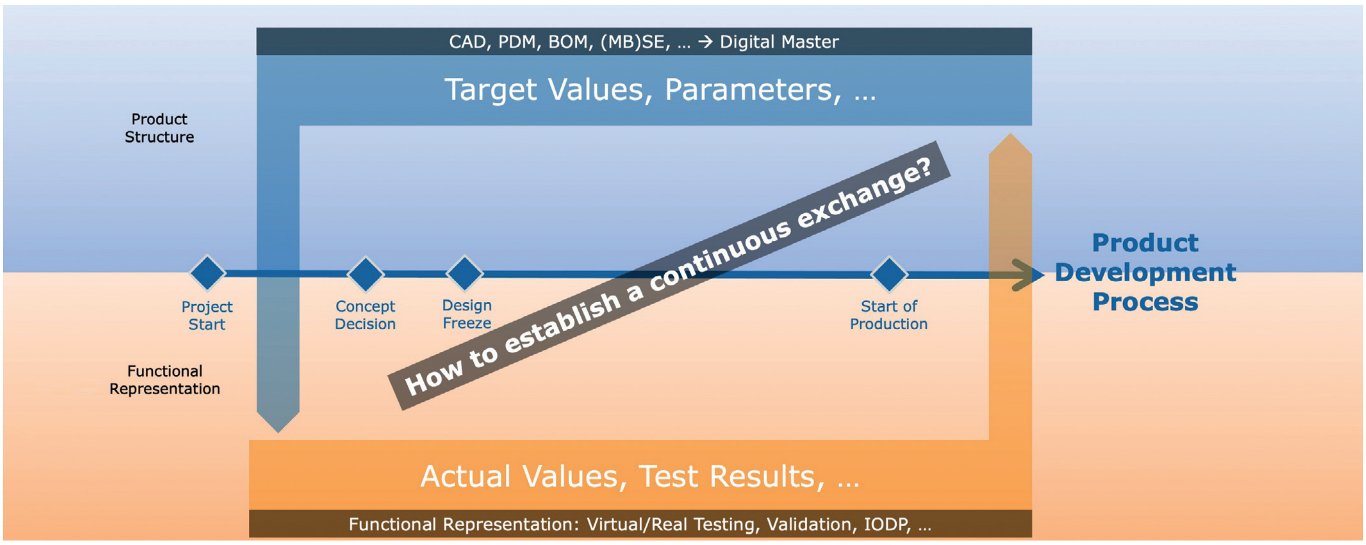

***Figure 1:** Two worlds: product structure and functional representation*

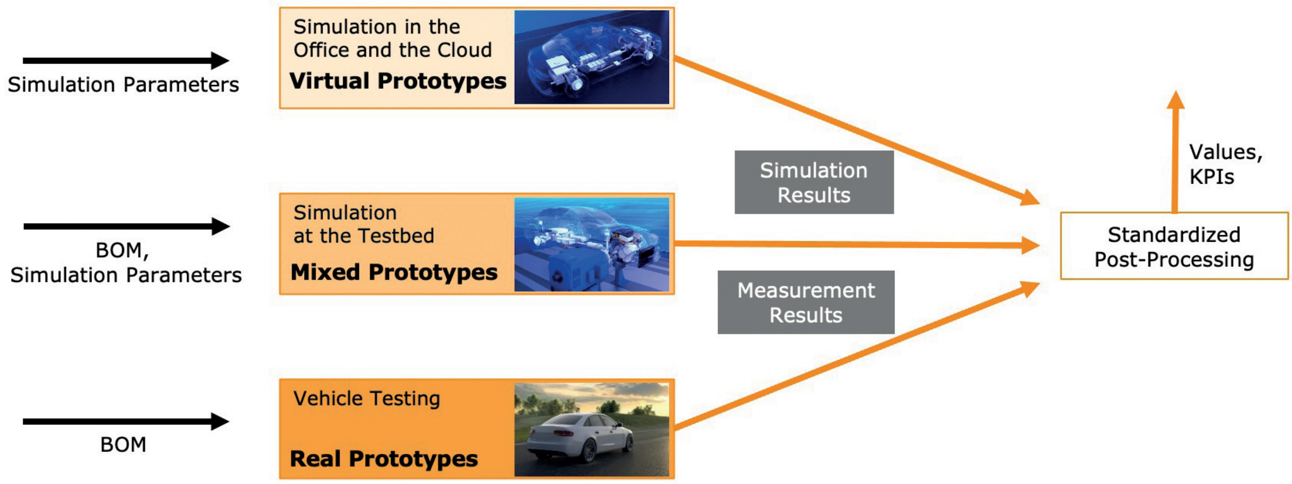

Figure 2: *Testing with virtual, mixed and real prototypes*

have been solved by linking simulation models as well as simulation and testbeds, but not all. One challenge has kept us very busy in recent years: How to get the right data to the right place at the right time?

Our vision of continuous V&V requires prototypes in all phases of the development process; hence we apply virtual, mixed or real prototypes (Figure 2). However, virtual and mixed prototypes are often domain-specific and not as complete as real prototypes. This massively limits the reusability of such prototypes.

A data model to address the interoperability challenge!

Before we go deeper, we want to define our "functional prototype". AVL's Functional Prototype Management allows continuous V&V at product-, subsystem- and component-level throughout the development process. A functional prototype (FPT) is a representation of the state of a product's functionality. Having heterogenic virtual and mixed prototypes in mind, a FPT has a focus on specific development activities, i.e.< a product can be represented by a set of FPTs. A FPT is valid throughout the overall development process (including the in-use phase).

Requirements and development activities

A new product is typically developed in a development project, e.g. when an automotive OEM develops a new car – from the start of design to the start of production (SOP), and nowadays even further extended to the customer's in-use phase. A customer buy a specific variant of the vehicle that is available e.g. in their local market. At the beginning of the development process, a (huge) set of requirements – such as fuel consumption, performance targets, driveability behavior or ADAS functionality – is defined. Each variant is described with corresponding target values (e.g. electric driving range 60 km), standardized test procedures (e.g. "in standard conditions") are determined.

The functional prototype

Throughout the project, we want to assess the risk of whether or not all target values will be met at SOP. Taking a look at the driving range of an electric vehicle, we have to compare the actual achievable value for the driving range to the originally defined target value. For this purpose, we perform a test using a prototype – virtual, mixed or real. A so-called "maturity level" indicates the difference between educated guesses based on experience (at the beginning of a project) or simulations in different complexity (throughout the process) vs. pre-series production cars (very late). In this sense, a low maturity level indicates a high risk that we will not reach the target value whereas a high maturity indicates high degree of certainty that the target value will be achieved.

Now let's look at the bigger picture: Wouldn't it be great to get all V&V information (actual and characteristic values, maturity levels, over time/throughout the development process) in a structured way for all our requirements – and for all variants?

The concept of Functional Prototype Management offers the solution for this. It retrieves the target values from the requirements and reports back the current status of fulfillment of the requirement, including information on the degree of maturity. In this way, we know at any point in the project whether a target has been met or not and what the risk is to not meet it at SOP.

From a practical point of view, our concept defines a set of functional prototypes which reflects the typical development activities, like performance & consumption, driveability or ADAS.

Sounds like a good concept, but what does it look like in reality?

Now, how can we continuously test our product if hardware-based prototypes are only available in later phases of the development process (Figure 2)?

In general, there are three options for testing a product:

- Virtual prototypes
- Mixed (virtual and real) prototypes
- Real prototypes

Let's start with the real prototype. Most of us are quite familiar with this type of testing. Testing with a real prototype requires that we build it first – based on a bill of materials (BOM). From on-road tests, we derive measurement data for which we then apply (standardized) post-processing. This is how we obtain our actual values for a functional prototype (Figure 3). Testing with a real prototype is a well-known and established process, but real prototypes can only be tested very late in the development process.

Using simulation in the office and in the cloud we can create fully virtual prototypes. Such a virtual prototype typically consists of simulation models and co-simulation architectures. Models and architectures offer the possibility of reusability – in different projects or different phases of the process. And if so, we have to think about parameterizing the models, i.e. establishing a link between the parameters and the current version of the product design, so that the models become a virtual prototype for our product. Models, architecture and parameters have to be managed to guarantee traceability. Once we have the models, parameters and virtual prototypes, we perform the defined test to obtain the simulation results. Applying a (standardized) post-processing, we again have our actual values for the functional prototype (Figure 3). In this case we obviously get them much earlier in the development process.

And in reality, there is no gap between virtual and real prototypes. Individual components are already available before the first real prototype is built. We can extend these components with simulation to so-called mixed prototypes. A mixed prototype – part virtual, part real – therefore uses hardware components on testbeds augmented with simulation. For this, we can re-use simulation parameters, models and virtual prototypes from previous development phases. After the test, we again apply (standardized) post-processing to the measurement data and report the actual values back to the functional prototype.

By using all three types of prototypes in this way, we obtain traceable and structured data. Functional Prototype Management allows the re-use of models, parameters and virtual prototypes. In this context, it supports the design validation planning (DVP) by promoting reusability across development projects and enabling more concrete planning (i.e. What can be re-used vs. what needs to be created for the upcoming tests?).

Establishing the digital thread and kickstarting data-driven engineering

By defining and applying standardized tests and post-processing, we generate comparable results (i.e., actual values and corresponding maturity levels) across the entire creation process. The resulting functional prototype shows the development progress independent of the development environments. Based on the tests performed, it is possible to exactly trace which prototype was used when and how it was configured by establishing the link from parameters to the design.

In addition, the approach allows the creation of prototypes that are designed for reusability. The reusability of prototypes also means that they can be created outside of the product development projects. This leads to a decoupling of certain information objects and processes from the concrete development project and creates a common and centrally coordinated data area for all projects where models, results, parameters, target values, actual values, virtual prototype architectures, DVPs, test scenarios, etc. are collected.

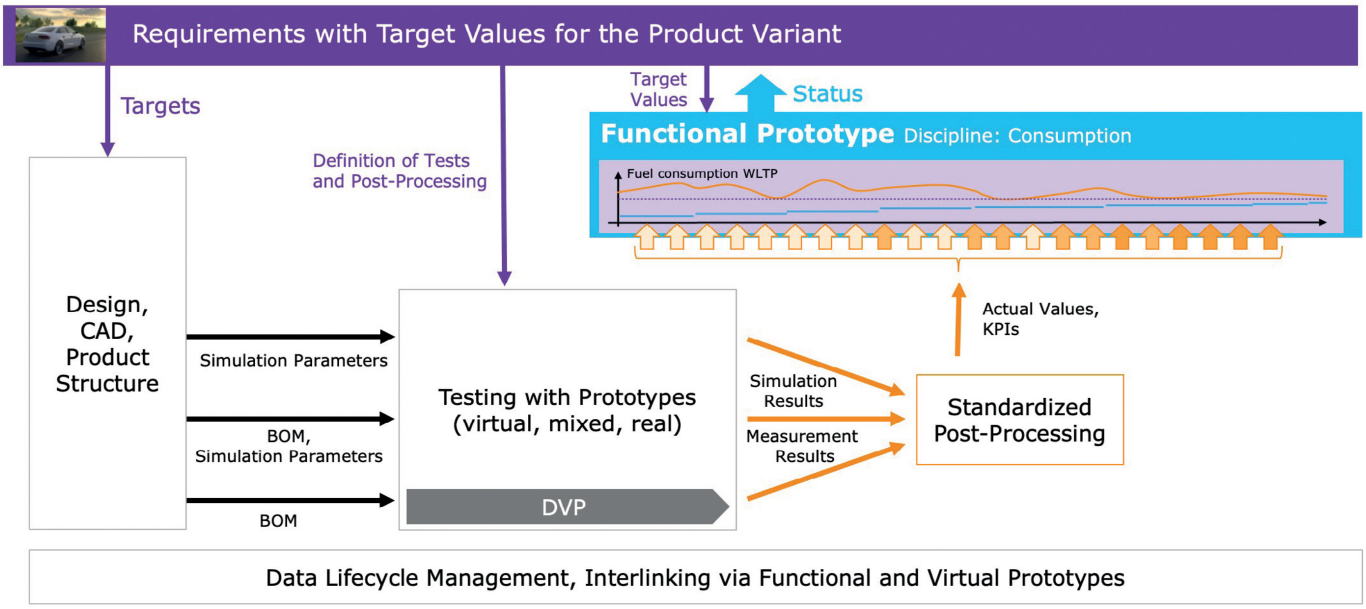

Figure 3: *Interlinking of product structure and functional representation through continuous validation*

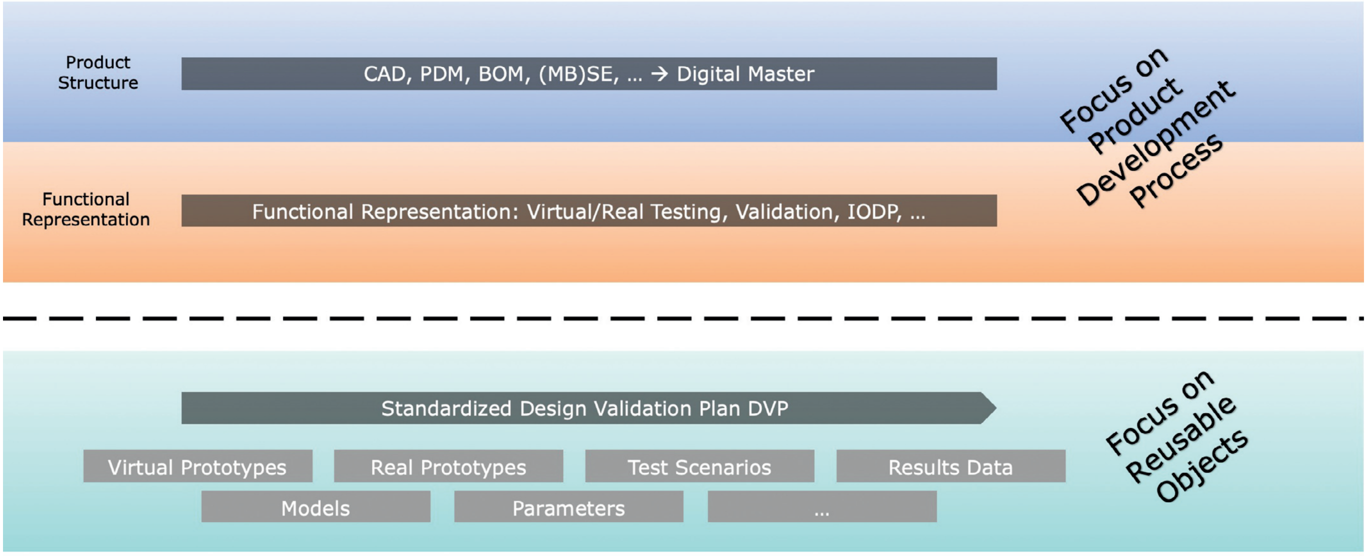

Figure 4: *: Distinction between product development process and environment provisioning*

In turn, we get the reusability of exactly these information objects across multiple development projects and this leads to development cost reduction and risk mitigation. With the context fully preserved, we establish the digital thread not only project by project, but across multiple projects (Figure 4).

In a nutshell

With neutral data integration, we can combine data from different sources and domains. The functional prototype is an advanced data model that provides a continuous and consistent view of the state of the product – across the entire life cycle from development to production and vehicle in-use. Functional Prototype Management provides one common concept for component-, subsystem- and product-level and enables transparency and monitoring for target achievement across the product creation process.

It connects the product structure-oriented world with the validation world. We can now apply automation and standardization to the validation within the development process, which in turn promotes data consistency. We are able to tell exactly which version of a model, parameter or prototype was used for a particular test. This makes lifecycle management of all relevant data entities by combining existing and new databases both possible and essential. A prerequisite for us is the identification of all relevant information objects. All this enables a deep understanding of data and the optimization of processes as well as methods based on massively structured data.

And your resulting benefits?

- Increased speed through efficient communication
- Increased consistency and traceability
- Increased process automation

If you want to tackle your digitalization challenge, get in contact with us. We analyze your current situation, derive together your "big picture" and specify concrete next steps.

Contact

Josef Zehetner
Chief Engineer
System Architecture IODP
AVL
josef.zehetner@avl.com

Contact

Katrin Moser
Marketing Specialist IODP
AVL
katrin.moser@avl.com