

## SOFTWARE-DEFINED VEHICLE

# Neue PLM-Anforderungen für die Fahrzeuge von morgen

New PLM demands for the cars of tomorrow



Immer mehr Fahrzeugfunktionen werden über Software definiert. Welche Anforderungen das Software-Defined Vehicle (SDV) an Entwicklungsprozesse und -werkzeuge stellt, erläutern Daniel Baldus, Principal und PLM-Berater bei der UNITY AG, und Dr. Patrick Müller, Chief Customer Officer bei CONTACT. Beide Unternehmen arbeiten in der Prozess- und Methodenberatung Hand in Hand. ■ An increasing number of vehicle functions are being defined by software. Daniel Baldus, Principal and PLM Consultant at UNITY AG, and Dr. Patrick Müller, Chief Customer Officer at CONTACT, explain what demands the Software-Defined Vehicle (SDV) places on development processes and tools. Both companies work hand in hand in process and methodology consulting.

**Software im Fahrzeug gibt es schon lange. Was macht ein Fahrzeug zum Software-Defined Vehicle?**

**Baldus:** Über 100 Jahre lang wurde der Automobilbau aus einer mechanischen Denkweise getrieben und die Software-Entwicklung dem untergeordnet. Jetzt wandelt sich die Philosophie grundlegend. Hersteller gehen davon aus, dass es schneller und kostengünstiger ist, das gesamte Fahrzeug so zu entwickeln wie Software.

**Müller:** Software kommt nicht mehr nur auf Komponentenebene, sondern als ganzheitliche Architektur zum Einsatz. Dazu kommt ein Prozess, der ein kontinuierliches Deployment ermöglicht, und die Notwendigkeit, jederzeit den Fahrzeugzustand inklusive der Software zu kennen oder erkennen zu können.

**Was bedeutet das für die PLM-Systemlandschaft?**

**Müller:** Zur klassischen Schichten-Architektur mit ERP-, PDM/PLM-, TDM- und Autorensystemen kommt womöglich eine weitere, explizite Schicht für das Konfigurations- und Change-Management. Daneben wird es ein Application Lifecycle Management (ALM) und Source Code Management (SCM) für die Verwaltung von Requirements und Soft-

ware-Code geben, die zu gewissen Anteilen auch im PLM verankert werden. Wichtig ist ein System, das den übergreifenden Reifegradprozess mit dem DevOps-Prozess für die Software-Entwicklung verbindet.

**Baldus:** PLM-Systeme werden sich weiterentwickeln müssen, um Daten aus Silos zu verknüpfen und vorausschauend auszuwerten. Sie werden sich viel stärker in Richtung End-to-End-Prozesse bewegen, um das Produkt auch in nachgelagerten Prozessen zu verfolgen. Dieses Produkt-Tracking gibt Aufschluss darüber, welche Software ich aufspielen und welche Services ich anbieten kann.

**Welchen Stellenwert hat das durchgängige Datenmanagement für das Thema SDV?**

**Baldus:** Auf der grünen Wiese würde sich das Thema Datendurchgängigkeit einigermaßen einfach darstellen. Das Problem sind die Altdaten, die sich über 30 Jahre in verschiedenen Silos aufgebaut haben und deren Stände untereinander inkonsistent sind.

**Müller:** Um das „aufzuräumen“, braucht es PLM-Technologie mit flexiblen Datenmodellen und Automatismen, die klassische PLM-Automatiken und Datenintegrität mit neuen Möglichkeiten der KI-Unterstützung verbinden. PLM-Technologie kann



Dr. Patrick Müller, Chief Customer Officer at CONTACT

sich aufgrund ihrer Vorzüge im Datenmodell im Kontext des SDV immer mehr zum zentralen Tool für die Datenorchestrierung entwickeln.

#### Wodurch unterscheiden sich Ansätze der verschiedenen PLM-Anbieter beim Thema SDV?

**Baldus:** Viele PLM-Anbieter versuchen, ihr Angebot in diese Richtung mit dem Zukauf von Auto-Systemen zu erweitern – können diese dann aber nur mühsam in eine durchgängige Toolchain integrieren. Außerdem haben die wenigsten PLM-Hersteller flexible Architekturen, die es erlauben, Software anders zu handhaben als Hardware. Das ist aber eine entscheidende Voraussetzung für das SDV.

**Müller:** Im Grunde sind diverse PDM-Systeme nicht ideal für die Verwaltung von Software-Kompatibilitäten ausgelegt. Allerdings eignen sich manche Systeme besser als andere. Unser Ziel ist es, Standard-Funktionen zu entwickeln, mit denen sich mechatronische Komponenten wie aus einem Baukasten in Baureihen integrieren und mit den entsprechenden Software-Ständen und Kompatibilitäten verbinden lassen. CONTACT Elements tritt dabei als PLM-Herausforderer an. Wir sind uns unserer Fähigkeiten und Differenzierungsmerkmale sicher.

#### Inwieweit lassen sich die Veränderungen im Zusammenhang mit dem SDV auch auf andere Branchen übertragen?

**Baldus:** Nicht die Branche ist entscheidend, sondern die Art der Produkte. Produkte mit Software-Anteil gibt es inzwischen in vielen Industrien – überall dort machen Lösungen wie in der Automobilindustrie Sinn. Eine große Chance für Unternehmen liegt auch darin, dass sich mechatronische Produkte viel standardisierter entwickeln und herstellen lassen als die rein mechanischen Produkte der Vergangenheit.

#### Unterscheidet sich das nicht vom CONTACT-Ansatz mit den industriellen Blaupausen für verschiedene Branchen?

**Müller:** Tatsächlich nicht. Bei den industriellen Blueprints geht es vielmehr um branchenspezifische Prozesse, etwa mit Blick auf spezielle Richtlinien. Gemeinsamkeiten ergeben sich aber in der Zusammenführung der Disziplinen. Da bin ich mit Daniel völlig einer Meinung: Die Denkweise des Software Defined Products gilt für die gesamte Fertigungsindustrie, auch wenn die Ausprägung für einen Fahrzeugbauer eine andere ist als für einen Anlagenbauer.

#### ■ Software in cars is nothing new. What makes a vehicle a Software-Defined Vehicle?

**Baldus:** For over 100 years, automotive engineering was driven by a mechanical mindset, with software development secondary. Today, the philosophy is fundamentally changing. Manufacturers now assume it is faster and more cost-effective to develop the entire vehicle in the same way as software.

**Müller:** Software is no longer just used at the component level, but as a holistic architecture. This includes a process that enables continuous deployment and the need to know or be able to recognize the vehicle's condition, including the software, at all times.

#### What does this mean for the PLM system landscape?

**Müller:** In addition to the traditional layered architecture with ERP, PDM/PLM, TDM, and authoring

systems, there may be another explicit layer for configuration and change management. There will also be Application Lifecycle Management (ALM) and Source Code Management (SCM) for managing requirements and software code, which will also be anchored in the PLM to a certain extent. It is important to have a system that links the overarching maturity level process with the DevOps process for software development.

**Baldus:** PLM systems will have to evolve to link data from silos and evaluate it predictively. They will move much more towards end-to-end processes to monitor the product even in downstream processes. This product tracking provides information about which software to install and which services to offer.

#### How important is consistent data management for the SDV?

**Baldus:** In a greenfield scenario, data consistency would be a reasonably simple issue. The problem is the legacy data that has built up over 30 years in different silos with statuses inconsistent with one another.

**Müller:** To “clean this up”, we need PLM technology with flexible data models and automatisms that combine classic PLM automation and data integrity with new possibilities for AI support. Due to its advantages in data modeling, PLM technology can increasingly develop into the central tool for data orchestration in the context of the SDV.

#### How do the approaches of different PLM vendors differ regarding the SDV?

**Baldus:** Many PLM vendors are trying to expand their offerings in this direction by acquiring authoring systems – but then struggle to integrate them into a seamless toolchain. Furthermore, few PLM manufacturers have flexible architectures that allow software to be handled differently from hardware. This, however, is a crucial requirement for the SDV.

**Müller:** Basically, many PDM systems are not ideally designed for managing software compatibilities. However, some systems are better suited than others. We aim to develop standard functions that al-

low mechatronic components to be integrated into model-series-like building blocks and linked to the corresponding software versions and compatibilities. CONTACT Elements enters as a PLM challenger in this arena. We are confident in our capabilities and differentiating features.

#### To what extent can the changes associated with the SDV be transferred to other industries?

**Baldus:** It's not the industry that matters, but the product type. Products with a software component are now present in many sectors – solutions like those in the automotive industry make sense everywhere. Another promising opportunity for companies is that mechatronic products can be developed and manufactured in a much more standardized way than the purely mechanical products of the past.

#### Doesn't that contradict CONTACT's approach regarding industrial blueprints for various industries?

**Müller:** Actually, no. Industrial blueprints are more about industry-specific processes, for example, concerning special guidelines. However, similarities arise in the convergence of disciplines. I completely agree with Daniel: the Software-Defined Product mindset applies to the entire manufacturing industry, even if the manifestation is different for an automotive company than for a plant manufacturer. ■

Daniel Baldus, Principal and PLM Consultant at UNITY AG

