ptc

DIGITAL TRANSFORMS PHYSICAL

Excellence in Automotive Software Engineering

Key Factors of Development Maturity to Accelerate Digital Mobility Innovation

Foreword

It's no secret that automakers have traditionally considered software to come in second place to hardware.

However, industry disruptions like autonomous driving, shared mobility, and electrification have completely turned that reality on its head. The automotive software market was valued at 12.5 billion U.S. dollars worldwide and is expected to reach a staggering **13.8 billion U.S. dollars by 2025**. Software has become the main value driver in automotive product portfolios; software is the value of the car now. In essence, the primary goal of automotive Original Equipment Manufacturers (OEMs) now is to become software-driven companies.

While some automotive companies have already embraced this disruption, many are still grappling with the shift. OEMs need to radically transform the way they think about and manage software development because traditional methods just won't cut it anymore.

According to **research conducted by McKinsey**, organizational software development maturity is today's key differentiator between innovative R&D organizations and those that will get left behind, with the former demonstrating 3-6 times more output and quality than the latter. ►

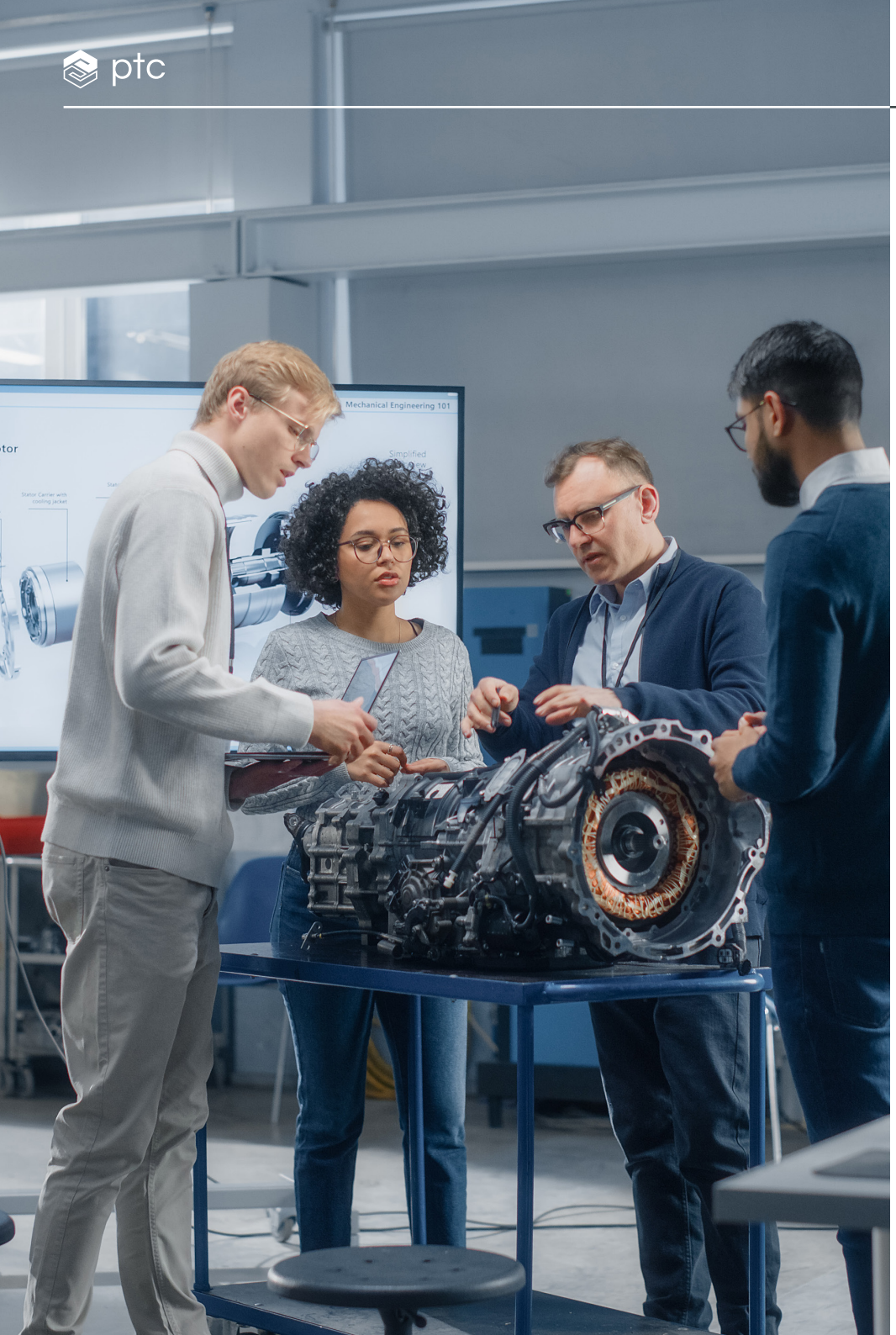
Contributing expert

About Dr. Siegmar Haasis

After 27 years of digitalization and leadership experience at Daimler (as CIO Asia/China and for 7 years as CIO R&D Mercedes-Benz worldwide), Dr. Siegmar Haasis started his own business in 2021 as a Digital Advisor.

Sigi advises OEMs and System Suppliers on Value-based Digital Transformation as well as Tech, Software, and IT Companies along a value-based go to market strategy.

Visit Sigi at **HaasisDEC.com** Digital Engineering Consulting.



Organizations that resist this transformation will face significant challenges in the future due to the growing complexity-productivity gap. Market demands and software complexity are skyrocketing while productivity levels cannot keep up.

Without transforming the way they approach the development of software, automakers risk:

- **Lagging innovation**
- **Wasting resources on software maintenance**
- **Production delays**
- **Going over budget in innovation-heavy product development**
- **Reputation-damaging recalls**
- **Health and safety risks**
- **Significant cybersecurity issues**
- **Missing out on new sources of revenue**
- **Lack of attraction for hiring talented software developers**

The true key to bridging the gap for mobility companies is optimizing software operating models and mastering excellence in automotive software by adapting the design, development, delivery, and management of software products. This eBook will provide practical insights into mastering the development of automotive software, with a focus on development models and the use of Agile to accelerate innovation.

The current landscape of automotive software

Nowadays, it's not just fancy new functions like adaptive cruise control and parking support that use software. Software is involved in almost all the functions of modern cars – from engine control to heating and cooling, power steering, transmission, parking assisting, locks, lighting, navigation, and even smart passenger seats, software is everywhere, and it's increasingly interconnected.

That's why getting a high-end car on the road these days **takes 100 million lines of code**, vs a Boeing 787's 15 million or Facebook's 62 million. Fully autonomous vehicles are estimated to need somewhere between 300-500 million lines of code, meaning that future cars will house some of the most impressive software systems we've ever seen in just a few years' time.

Why traditional software development isn't enough anymore

In the past, vehicle software was largely developed externally and delivered to the OEM by ECU suppliers. Close coordination throughout the product development process ensured that the software worked well with the specific hardware it was embedded in.

Increasingly, OEMs strive to develop large parts of the software themselves, based on a domain-specific vehicle reference architecture or platform. In addition to shorter development times, this will enable OEMs to continuously flash the new software into the vehicles via OTA, keeping their functionality and behavior up to date and establishing new revenue streams.

All these ways software is used by modern cars mean skyrocketing complexity in the functional and architectural levels of mobility product development. Productivity can't keep pace with mounting complexity, resulting in insufficient software capabilities and limiting the profitability of carmakers.

That handicap, and the widening gap between complexity and productivity, is likely to grow in significance as software becomes a key driving force across the various tiers of automotive innovation. Mastering the delivery of automotive software is becoming a vital strategic concern for developers of automotive products. The higher up they are in the Tier system, the sooner they are likely to feel the effects of this digital challenge.

Key aspects of responding to the digital complexity challenge

To navigate these challenges, automotive developers need to innovate new operating models that enable them to reduce complexity while increasing efficiency in software development. According to McKinsey research and expert analysis, this new model should span the following aspects:

1. The software product's design & architecture
2. Managing software requirements
3. The development of digital automotive products (including methodologies for optimized & efficient software development)
4. The organization's enablement to boost innovation

That's why getting a high-end car on the road these days takes 100 million lines of code, vs a Boeing 787's 15 million or Facebook's 62 million. Fully autonomous vehicles are estimated to need somewhere between 300-500 million lines of code, meaning that future cars will house some of the most impressive software systems we've ever seen in just a few years' time.

1. Modernizing product design & architecture

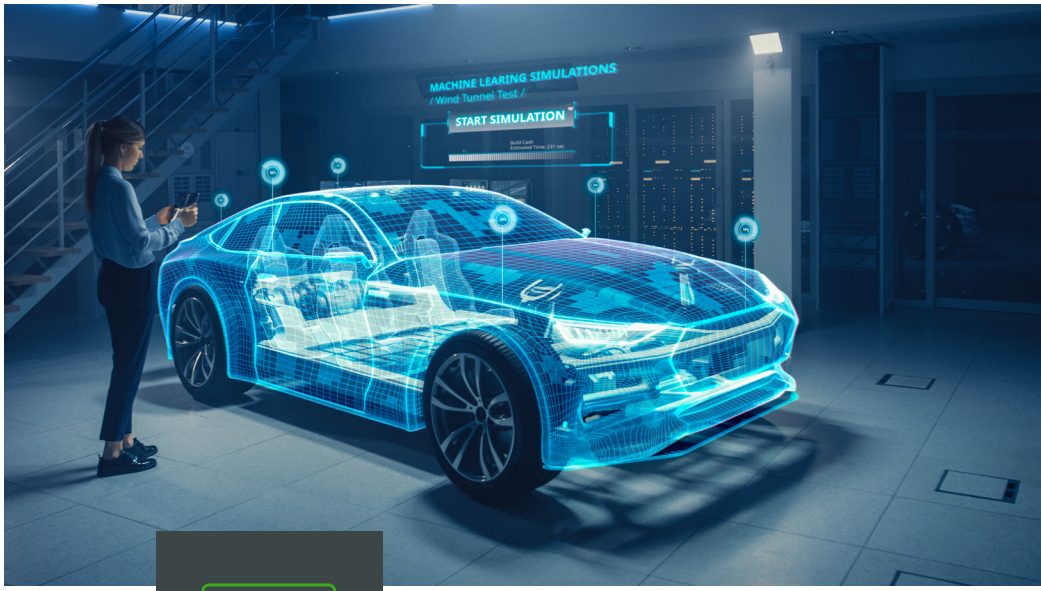
Modularization and software-hardware decoupling, combined with a Systems Engineering approach, help accelerate the development of system-of-systems products.

First and foremost, automotive systems engineers should strive to decouple the hardware and software architectures that are the building blocks of their mobility products as much as possible.

Software and hardware components have very different lifecycles, with hardware development characterized by much longer cycle times. If the two are bound together, that limits the pace of innovation as software and hardware updates need to happen in lockstep. By decoupling hardware and software, carmakers can accelerate innovation and prepare for future tech trends in the digital mobility industry.

Decoupling also enables developers to move some heavy lifting from embedded computer systems to the cloud, reducing the need for powerful hardware built into cars.

In a decoupled setting, it's much easier to establish a modular approach to product development. With modular system architectures, software complexity becomes easier to manage, with clear boundaries between modules defining the requirements for each team or supplier working on each subcomponent. This greatly supports transparency and process visibility in the management of complex automotive product development projects. ►



Besides reducing architecture complexity, modularity also helps implement standardized operating systems for quality control, interoperability, and the harmonization of components across the various subsystems of automotive end products.

Thus, modularization supports the implementation of efficient reuse and Product Line Engineering strategies, while also enabling the easier management of interdependencies.

Controlling the impact of changes and tracing errors become easier, which helps fix emerging issues faster, accelerating cycle times and reducing costs.

In a decoupled development setting, it's important to define individual backlogs for each component and to establish clarity on when and how synchronization should happen across the realms of software and hardware.

At these integration milestones, auto developers should rely heavily on automated testing to capture issues on the fly. After a technical assessment by the experts, cross-functional systems are defined and organizationally corresponding roles of system owners are established

One solution that supports the adoption of a modularized approach is the implementation of the Systems Engineering concept. SE provides a holistic, all-encompassing framework to tackle the challenges caused by gaps and information silos in the product lifecycle, disconnected toolchains, product complexity, and misalignment between parallel development streams.

Automotive innovators are increasingly engaging with the concept of Systems Engineering as it helps handle complexity through the digital description of end-to-end system functions and interdependencies.

2. Mastering software requirements management

Transforming established requirements management processes to take a user-first perspective is vital for success.

While managing requirements is traditionally a strong suite of OEMs, much of that experience is only relevant in the hardware realm. Building out the capability to handle the complexity that software requirements are reaching in modern automotive systems is becoming a key factor of competitive advantage for carmakers. To that end, automotive innovators should look to digital-native companies, borrowing established methods and know-how.

As a key best practice, prioritizing customer value and treating these user-facing specifications separately from technical enabler requirements helps stay focused on delivering value for the end-user. Continuously updating requirements in a development process driven by customer feedback is inevitable (see more on this in the following section on Agile development).

End-to-end traceability is becoming a prerequisite for efficiency in this multi-layered development environment. Establishing a digital thread that connects all the phases of the lifecycle from requirements definition all the way to final acceptance is crucial. In the future, the development process will end at the end of the vehicle lifecycle rather than at the start of production, as software will continue to be updated into vehicles in the field via OTA. The gapless interlinking of work items and processes provides the clarity that's needed to navigate functional safety, user requirements, and rapid code changes against the backdrop of system-of-systems complexity.

In most cases, a modernization of software development tooling will be necessary to enable the development organization's new approach to requirements management. Highly integrated tools (whether they are organically integrated tool suites or purpose-built tools connected via interfaces) can deliver the robust visibility and change control that enables teams to maintain clarity in this accelerated development setting.

3. Adopting new methods of product development

Transforming established requirements management processes to take a user-first perspective is vital for success.

While managing requirements is traditionally a strong suite of OEMs, much of that experience is only relevant in the hardware realm. Building out the capability to handle the complexity that software requirements are reaching in modern automotive systems is becoming a key factor of competitive advantage for carmakers. To that end, automotive innovators should look to digital-native companies, borrowing established methods and know-how.

As a key best practice, prioritizing customer value and treating these user-facing specifications separately from technical enabler requirements helps stay focused on delivering value for the end-user. Continuously updating requirements in a development process driven by customer feedback is inevitable (see more on this in the following section on Agile development). End-to-end traceability is becoming a prerequisite for efficiency in this multi-layered development environment. Establishing a digital thread that connects all the phases of the lifecycle from requirements definition all the way to final acceptance is crucial. In the future, the development process will end at the end of the vehicle lifecycle rather than at the start of production, as software will continue to be updated into vehicles in the field via OTA. The gapless interlinking of work items and processes provides the clarity that's needed to navigate functional safety, user requirements, and rapid code changes against the backdrop of system-of-systems complexity.

In most cases, a modernization of software development tooling will be necessary to enable the development organization's new approach to requirements management. Highly integrated tools (whether they are organically integrated tool suites or purpose-built tools connected via interfaces) can deliver the robust visibility and change control that enables teams to maintain clarity in this accelerated development setting.

4. Enabling teams with future-proof tooling

Transforming established requirements management processes to take a user-first perspective is vital for success.

Modern software development tooling is a key enabler of user-focused and software-driven strategies and efficiency in automotive systems innovation.

Adopting all the above strategies and practices requires automotive systems development teams to operate in an entirely new way. Equipping the organization with an adequate tool infrastructure is invaluable in ensuring the success of this modern approach to automotive software development.

Managing high-velocity development processes while maintaining quality control, traceability, and transparency is greatly supported by integrated software suites that help teams manage the application development lifecycle.

Integrating data and processes is crucial for ensuring traceability and workflow automation. Companies have two options:

- **Building a best-of-breed toolset with interfaces between tools**
- **Using organically integrated and docking further software platforms into this central development hub**

Integrated suites such as cutting-edge Application Lifecycle Management platforms deliver a holistic view of software development and provide data access for stakeholders and specified tools covering all phases of the lifecycle. The use of such tools enables teams to automate workflows that support development, testing, and regulatory validation.

Overall, according to McKinsey's research, introducing a standardized toolchain can help improve productivity by 30-40% by enabling the use of Agile methods and automated testing procedures.

ALM tooling for Agile software engineering maturity

Using an Integrated, end-to-end Application Lifecycle Management (ALM) platform is key for transforming automotive software engineering capabilities. When implemented correctly, ALM platforms help OEMs ensure product quality, safety, and compliance. Using modern ALM tooling also promotes smooth collaboration and process improvement across the board.

Here are some of the ways that using an ALM platform supports automotive Agile transformation journeys:

1. Link requirements to business strategy and customer value

Excellent software is created by continuously adjusting requirements based on stakeholder feedback as you go along. Using Agile feedback loops and an ALM platform that links all the requirements to related artifacts, you can track changes and progress to make sure you're on track throughout the software development process.

2. Encourage effective internal and external collaboration

Whether it's cross-functional collaboration or working with external partners that you're looking to streamline, having all your tasks, documentation, requirements, and knowledge in the same platform is a huge help. This is also crucial for security reasons when it comes to collaborating with external partners. The support of international standards for data usage and exchange, such as ReqIF or SysML, also accelerates internal and external collaboration.

3. Easy overview of QA and testing processes

The more complex a software product is, the more complicated it is to QA and test it thoroughly in order to ensure the safest possible product as well as demonstrate compliance with industry regulations. In an ALM platform, you can get a birds-eye view of the testing process, tracking all the work items throughout the full product lifecycle and having test case statuses at hand.

4. End-to-end traceability for change management and compliance

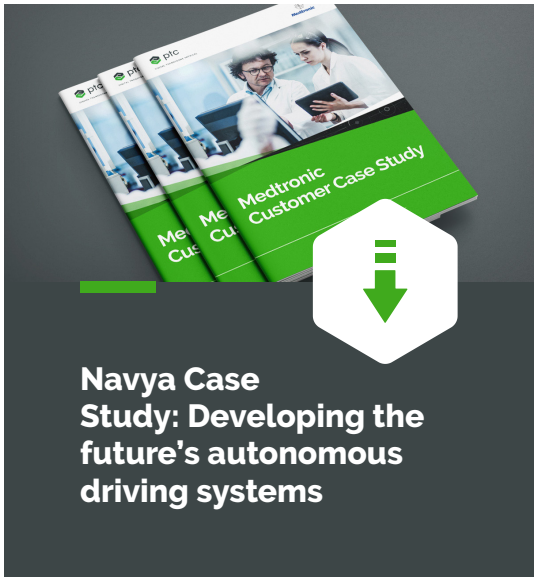
All the way from definition to acceptance, ALM platforms help companies trace every single requirement and change made throughout a software development project. This overview helps keep everyone involved on the same page and ensures a constant focus on user needs, all while making regulatory compliance a much smoother process.

Overall, integrated ALM helps manage risk at speed in automotive software development while providing a central collaboration platform and a single source of truth for all stakeholders.

Summary

Digital complexity not only means a momentous challenge for automotive product innovators. Successfully adapting to a software-driven market landscape also represents a decisive opportunity to establish a competitive advantage in the coming years and decades.

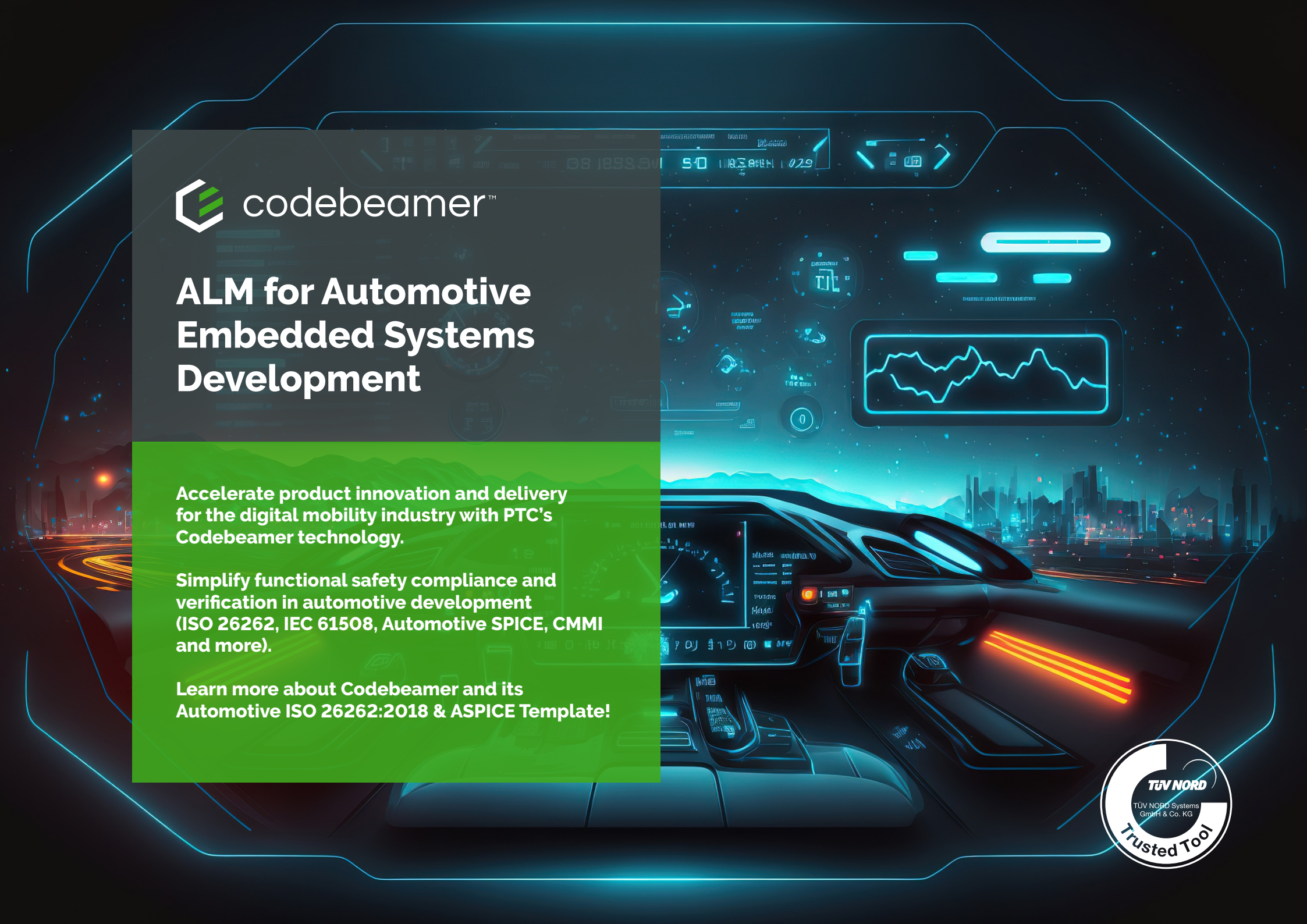
Drawing on the established strategies and best practices already used by technology companies in other industries, automotive developers can build out state-of-the-art software capabilities. By modernizing outdated engineering approaches, culture and mindset, processes, and tools, they can become user-focused digital mobility innovators, expanding market share and opening up new market opportunities and revenue streams. Missing out could mean being left behind as software increasingly redefines the automotive industry.



Sources: 9 automotive industry trends in 2022 and beyond / An Agile Game Plan for Automakers / Automotive Connectivity is Driving the Future of Mobility / Automotive software and electronics 2030: Mapping the sector's future landscape / Computer Chips inside Cars / From no mobility to future mobility: Where COVID-19 has accelerated change / How the Automotive Industry Can Benefit From Agile Software Development Introduction to the combined Application of Agile & Safety in Automotive Software Development / Software-Defined Vehicles – A Forthcoming Industrial Evolution / Software, Not Hardware, Will Drive the Future of Automotive / Upstream Security's 2020 Global Automotive Cybersecurity Report / When code is king: Mastering automotive software excellence / Why electric cars will take over sooner than you think

codebeamer™

ALM for Automotive Embedded Systems Development



Accelerate product innovation and delivery for the digital mobility industry with PTC's Codebeamer technology.

Simplify functional safety compliance and verification in automotive development (ISO 26262, IEC 61508, Automotive SPICE, CMMI and more).

Learn more about Codebeamer and its Automotive ISO 26262:2018 & ASPICE Template!

TÜV NORD
TÜV NORD Systems GmbH & Co. KG
Trusted Tool

DIGITAL TRANSFORMS PHYSICAL

121 Seaport Blvd, Boston, MA 02210 : ptc.com

018-excellence-in-automotive-software-engineering-02-09