

软件定义汽车时代： 领导者如何构建战略优势

成熟整车厂转型的战略指南：
驾驭软件复杂性，引领实现软件优先转型



前言

本战略指南旨在介绍如何将您眼前最大的难题，即软件复杂性，转化为您最强的竞争优势。

作为传统整车厂的领导者，您面临的核心挑战已不再是追赶新入局者，而是如何驾驭产品的决定性因素——巨大的软件复杂性。

转型已不是“是否进行”的疑问，而是“如何推进”的课题。本战略指南提供了构建必要能力的路线图，助您在新的行业格局中取胜。

指南定位

这是一套战略框架，助力传统整车厂转型为软件优先企业。您无需放弃既有优势，只需将曾经变革制造业时使用的卓越工程理念，应用于软件开发领域即可。

核心能力培养

直面核心挑战

了解“一亿行代码技术负债”为何成为影响当前企业利润率、创新速度与竞争地位的关键变量，以及需要采取哪些战略方法应对。

掌握平台工程

学习如何将软件开发从碎片化项目重组为一体化。您将了解标准化为何能加速开发，以及如何通过黄金路径和集成工作流实现质量默认达标。

构建软件基础

从架构验证到自动化合规，理清安全的软件定义汽车(SDV)开发需要哪些技术栈。

面向未来机遇的转型路线图

明悉平台所有权如何帮助您满足不断变化的市场需求(如人工智能安全合规)。获取四步实现组织转型的专家指南，这些经过验证的策略已为成功转型的整车厂带来回报。

目标受众

本指南适用于认识到渐进式改革无法弥补软件差距的高级技术专家与工程领导者，也适用于准备系统性改变组织内部软件理解、构建和交付方式的决策者。

通过战略转型，在原有能力基础上实现突破，加入领军者行列。汽车行业已发生巨大变化。本战略指南将指引您成为颠覆者。

核心术语

在深入探讨之前，我们先明确软件定义汽车转型所涉及的几个关键概念：

软件定义汽车 (SDV)

一种核心功能、特性和性能主要由软件决定的车辆，其相关功能不依赖于固定硬件，并能**通过软件持续得到增强**。该车辆相当于“带轮子的计算机”，而非“带计算机的汽车。”

平台所有权

对决定车辆功能、用户体验及未来能力的**核心软件层的掌控权**。平台所有权是一个谱系概念，因此您无需自主构建所有组件；此外，平台所有权还包含架构层面的控制权，使组织能够独立于供应商集成、修改和改进软件栈。

平台工程

一种**将所有工具链和工作流整合为“黄金路径”**的战略软件开发方法，使开发者无需从零搭建必要的基础设施即可自助完成开发。该方法将组织内部开发的基础设施视作产品进行管理。

工程平台

通过平台工程构建的具体解决方案，即**一套统一的工具、工作流和标准，可用于加速软件开发**。请不要将其与运行应用程序的硬件/软件平台混淆。

技术负债

代码中累积的复杂性和捷径，会提高后续更改的难度和成本。在汽车行业，该问题具体表现为，中间件中存在数百万行遗留的C/C++代码，难以进行合规和安全监管。

架构验证

持续检查软件实现是否符合预期设计的实践，可防止“架构侵蚀”，从而避免由此产生的集成问题、安全和合规风险。

引言

整车厂皆为软件公司

成功转型为软件公司的整车厂，将主导未来十年移动出行的格局；而犹豫不决者终将向果断行动的竞争对手寻求技术授权。

因此，问题的关键在于：您究竟想让谁掌握汽车的未来——是您的工程师团队，还是日益复杂的代码库？

现代汽车搭载的代码超过1.5亿行，预计未来十年这一数据还将增长十倍。然而，NASA的阿波罗11号仅需约14.5万行代码。如今，汽车的软件复杂性已远超将人类送上太空的火箭系统。

这一事实为汽车行业带来了深远影响。当**特斯拉、比亚迪、吉利**等新兴整车厂将汽车视为可快速迭代能力的软件平台时，传统整车厂还在为遗留架构和复杂供应链所困。

平台所有权 = 掌控产品的未来

平台所有权是一个谱系概念。其核心不在于自主构建所有组件，而是掌控决定客户体验和推动创新的关键层级。即便组件来自不同的供应商，只要掌控着架构与集成，您仍能“拥有自己的平台”，并在此基础上构建独特的用户体验。当您掌控了系统的连接方式与升级路径，您便掌握了竞争的主动权。

新的竞争格局由一个共同的指标决定： 快速交付高质量软件的能力

2024年，**Recall Masters**报告显示，超过2800万辆汽车受到召回影响，其中174起召回事件源于软件与电子系统故障，涉及1380万辆汽车。¹

软件发布后，修复缺陷的费用可能是开发阶段的百倍还不止，因此，软件质量已不再只是工程问题，更是领导层所面临的业务风险。

问题不在于这场转型是否正在发生，而在于您是选择掌控它，还是任其裹挟。本指南提供了战略路线图，可助您重掌局面，将潜在负债转化为竞争优势。

¹Recall Masters 《2024年度召回情况报告》 ↗



1

传统方法 在SDV时代难以为继

传统方法 在SDV时代难以为继

这些年来，您早已听闻：传统方法已无法维系。行业领导者早已认识到，旧工作模式无法适应软件定义时代。

但认识到问题并不等于解决问题。许多传统整车厂仍然保持着旧工作模式，这种模式持续阻碍创新，它们也因此被称为“一亿行代码技术负债”（100M-Line Tech Debt）。

存在这样一个顽固的挑战，会使每个新项目都更加费时、费力、费钱。它不仅令技术复杂性逐年累积，还会让开发团队每日被挫败感折磨。

这类技术负债根源于传统汽车架构的碎片化基础。例如，开发团队需要面对不同供应商提供的数十个独立割裂的电子控制单元(ECU)，其构成的基础架构沉重不堪。这种碎片化的形式基本不可能实现全车范围的软件升级(SDV的核心能力)。

并且，这会引发一系列连锁问题：

“核心问题在于遗留架构。一辆车大约包含60-90个独立的电子控制单元；每个单元都是来自不同一级供应商的黑箱，需要通过速度慢、带宽低的网络进行通信。

对于整车厂而言，这种碎片化模式基本无法构建快速覆盖整车范围的固件在线升级(FOTA)和软件在线升级(SOTA)，而这正是现代SDV制造商的优势所在。

于是，这就形成了恶性循环：每当需要新增功能时，默认直接增加独立的ECU；而每增加一个单元，代码库就会变得更庞大、更复杂，每代车型的更新难度也随之增加。”



罗淼

技术战略总监

Qt Group

成本攀升

如今，软件问题已成为车辆被大量召回的主要原因，其中的碎片化架构不仅增加了问题发生概率，还大幅提高了修复成本。2024年全球汽车网络安全市场规模为35亿美元，预计2025至2034年间将以11.6%的复合年增长率持续增长。²

阻碍未来收益

架构掌控薄弱会导致部署可靠的空中下载(OTA)更新异常复杂。分析机构TechInsights汽车业务副总裁Ian Riches指出，**特斯拉**在2012年率先推出了OTA更新，但13年后的今天，这项技术“对于大多数车企仍然属于全新领域”。³

²GM Insights 《汽车网络安全市场报告》 ↗

³MotorTrend 《汽车OTA更新推进缓慢》 ↗

“
软件问题现已成为
车辆召回的主要原因。
”

拖累创新速度

技术负债会形成“创新阻力”，未记录的依赖关系和集成阻碍基础更新。团队处理遗留问题耗费的时间远超创造客户价值的时间。这会形成巨大的竞争差距：SDV可在数周内发布更新，而传统团队则需6-12个月才能完成相同变更。这种迟缓不仅会带来不便，还会上升为安全风险。当出现关键漏洞时，敏捷的竞争对手可在数天内通过FOTA修复所有车辆，而碎片化架构却让客户暴露在风险下达数月之久。

如果缺乏清晰且管理得当的架构，整车厂将无法接入SDV时代的决定性产物——基于服务的营收模式。



行业经验

每增加一个新项目，技术负债都会指数级增加，团队尝试集成现代系统（如ADAS和AI功能）和遗留中间件时遇到的挑战也会成倍增长。面对此状，具有前瞻思维的整车厂会重新评估先前的战略。

例如，**福特**做出战略调整，将软件研发重心转向其先进的电气架构。这表明业界已逐渐认识到，要想实现可持续创新，就必须先夯实基础平台。⁴ 这一转变也展现了传统整车厂如何在利用自身优势和数十年工程经验的同时，对软件基础进行战略性现代化改造。

⁴ CNBC 《福特终止开发类特斯拉电子大脑项目》 [↗](#)

```
import Quick3DAssets.LaneAssist 1.0
import Quick3DAssets.DangerArrow 1.0
import QtQuick.Timeline
import Data 1.0 as Data

// Turn signal state
property bool signaling: laneAssist.lane < 0
property bool turning: "Turning" === state

property alias lanelines: lanelines
property alias timelineAnimationAdas: timelineAnimationAdas

environment: sceneEnvironment

// Lighting modes
Keys.onDigit1Pressed: dynamicLighting.state = ""
Keys.onDigit2Pressed: dynamicLighting.state = "Day"
Keys.onDigit3Pressed: dynamicLighting.state = "Dusk"

// Weather modes
Keys.onDigit4Pressed: currentWeather = 0
Keys.onDigit5Pressed: currentWeather = 1
Keys.onDigit6Pressed: currentWeather = 2
Keys.onDigit7Pressed: currentWeather = 3
```

2

硬件企业的 软件工程规范

硬件企业的软件工程规范

如今，整车厂仍在使用硬件时代的流程艰难参与竞争。随着软件成为主要的价值驱动因素，他们必须全面重组开发战略，并在此基础上构建新的规范。

数十年来，传统整车厂一直在完善制造领域的规范。而软件优先公司也在代码中运用了这种严谨的思维方式，将质量与架构视为统一且持续的过程。

这一转变至关重要，因为软件运行遵循的三大铁律与硬件截然不同：

- 软件持续变化
- 软件问题成倍增长，而非逐个增加
- 软件可以持续优化——前提是您能掌控它

面对这些系统性挑战，单独的工具已无法应对。要摆脱这些挑战的束缚，必须战略性采用面向构建新常态的规范，即平台工程。



平台工程何以成功

您可以将平台工程理解为，把车辆装配线的应用规范，照搬到您的新型软件工厂中。

正如实体装配线依赖于标准化工具和可重复流程，工程平台可为开发者提供标准化的工具、模板、API和最佳实践。它们的目的相同，都是为了简化开发流程、降低复杂度并提升生产力。工程平台能够提供可靠、一致的环境，以便团队快速构建创新功能，无需每次从零开始搭建基础设施。

这是一套规范，而非单一的工具或技术。正如精益制造改变了汽车制造方式，平台工程如今也正在重塑软件的构建方式。

[了解更多关于平台工程的基础知识](#) ↗

数据为证

平台工程战略是应对现代汽车软件开发挑战最有效的战略。为提供数据支撑，Qt Group委托Forrester Consulting于2024年开展市场调研，证实了这一转型的重大商业价值。此次调研走访了300多位嵌入式软件负责人与专业人士，揭示了以下核心优势：

- **开发效率提升59%**
大部分受访者都认为这是最重大的运营改善。
- **增强合规性与安全性**
既能满足严格监管要求，又能保持稳健的安全措施。
- **降低维护成本**
降低软件平台的长期拥有成本。

调研还特别指出，对于汽车行业而言，采用平台工程战略最大的好处在于：嵌入式软件开发团队（包括开发者、设计师和测试人员）之间的协作效率显著提升。

综合来看，这些数据表明平台战略能够提供显著的竞争优势，并帮助企业更快、更安全地进行创新。

平台工程的交付价值

平台工程与传统方法不同，传统方法将质量、开发和部署视作独立的环节，而前者则通过统一的工程平台构建协同的方法，其具体特征如下：

标准化不意味着僵化：

支持团队在80%的用例中遵循“黄金路径”，同时保持创新所需的灵活性

质量保证贯穿始终：

采用端到端的软件开发解决方案，其中包含在开发工作流程中持续保证合规性的工具

保持架构整洁：

提供工具持续验证实现是否符合设计，避免因软件侵蚀产生技术债务

可复用性成为默认选项：

一次构建出的组件可用于多个项目，35%的受访者表示上线速度有所提高

从碎片化工具转向集成平台，能够有效杜绝绕过质量和架构标准的现象，从而根治“一亿行代码技术债务”问题。

如需深入了解平台工程的原则，
请参阅[《铺就黄金路径：高管与工程师如何协同推进平台工程》](#) ↗

3

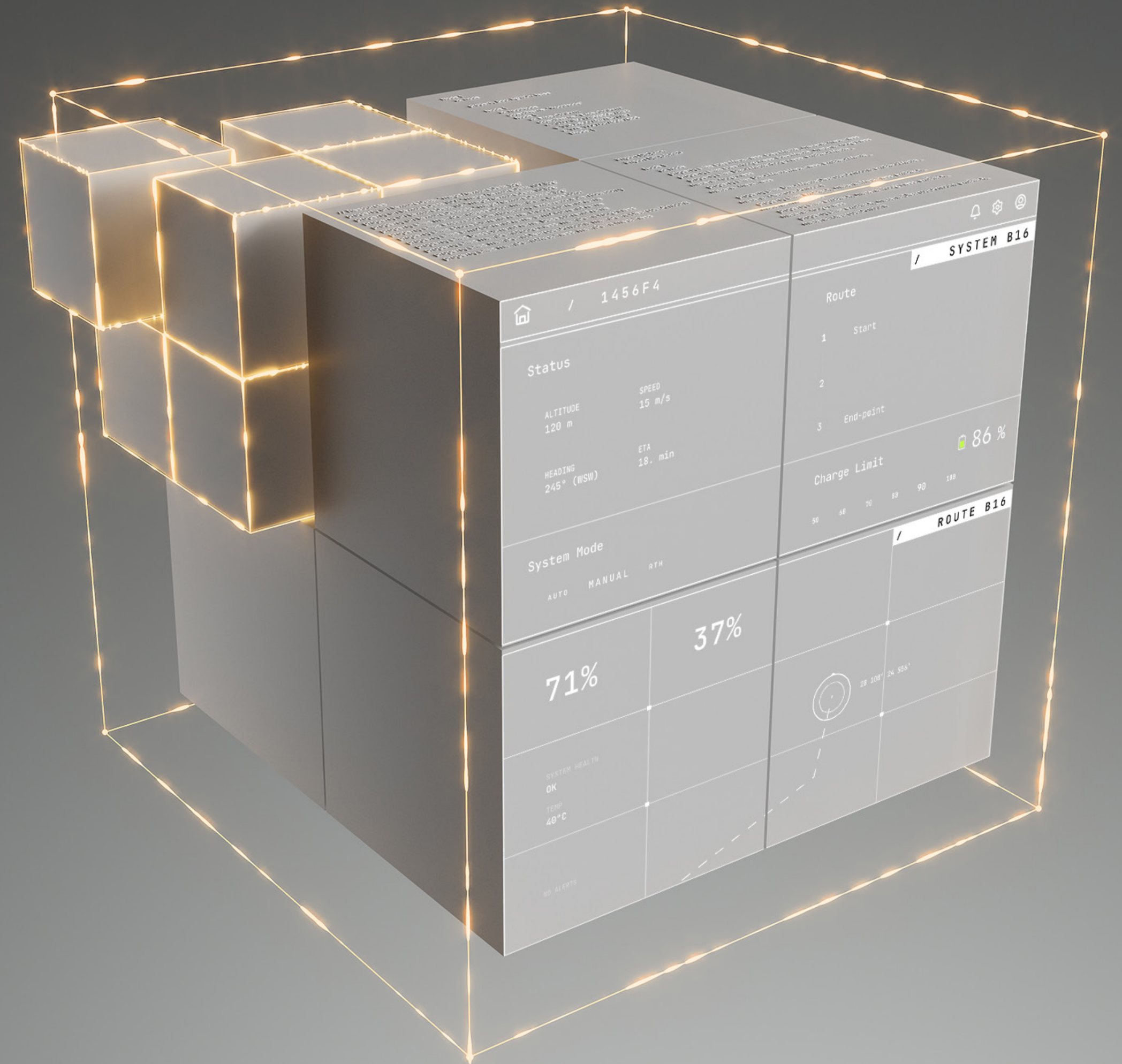
质量、安全与保障 构筑基石

质量、安全与保障 构筑基石

平台工程将质量、安全和保障贯穿整个开发生命周期，充分体现了软件优先的理念。该方法对SDV至关重要，能助力进一步掌控平台。

质量始于架构

构建高效平台不仅需要优质代码，还需要端到端解决方案，将软件开发与质量保证整合为持续流程。这一切始于架构完整性，依托一整套集成的现代化工具实现。



构建安全开发环境所需的完整技术栈

开发现代SDV所需的安全平台主要依靠两大支柱：稳健的开发工具链和无缝集成的质量保证解决方案。

开发工具链

- **单元测试框架**，用于在持续集成中快速运行单元测试
- **性能分析工具**，用于优化用户界面响应速度与流畅度
- **静态构建方案**，面向高完整性、安全关键型系统，满足其对更高可控性、安全性及可认证性的严格要求
- 详尽的**软件物料清单**，用于检查软件依赖项，在出现漏洞时迅速采取措施
- **长期保障**，保证API长年稳定

集成质量保证

- 防止设计漂移的**架构验证**
- **嵌入式合规检查** (涵盖MISRA、ISO 26262、AUTOSAR等标准)
- **全面的代码覆盖率分析** (支持MC/DC)
- 跨GUI、静态分析和性能的**集成测试**

高质量带来的复利回报

当质量作为开发流程的一部分，而不是最终附加环节时，事半功倍：

- 缺陷可在早期被发现，修复代价远小于后期
- 自动生成合规文档，无需手动审计
- 开发者可获得即时反馈，无需等待漫长的审查周期
- 架构清晰且易于维护

在推进平台工程的转型过程中，**43%*的组织认为安全与合规是最大的挑战。**这两点对于安全至上的汽车应用尤为重要，在此类关键任务与安全关键型应用中，开发团队需要对完整生命周期全权负责，从代码敲下第一行开始，直至设备最终退役。

更多关于汽车软件测试最佳实践的信息，
请参阅 [《汽车软件测试：SDV时代的质量保障之道》](#)。

从基石到创新：实践成果

转型为软件优先企业不仅需要新工具，还需要转变思维方式。过去，汽车由硬件定义，软件仅起辅助作用；但现在，汽车由软件定义，而硬件仅作支撑。

进行转型时，组织必须构建以软件设计、构建和测试为核心驱动规范。让这些规范落地生根、得以细化，不再是空谈理论，而是经过了行业领军者的实践检验。

在2025年Qt全球峰会上，**哈曼国际**展示了其如何通过自主掌控软件技术栈，在确保品牌所要求的质量与安全水准的同时，成功打造出差异化的用户体验。

[观看哈曼案例分享的完整视频](#) ↗

该案例很好地展示了汽车制造商如何通过掌控自有平台(包括UI框架、渲染引擎等)获取以下关键优势：

- **无损定制：**在保持安全标准的同时，针对特定硬件定制框架
- **面向未来的架构：**掌控长期发展，避免供应商束缚
- **无缝集成：**融合自有创新与合作伙伴解决方案
- **数据驱动优化：**基于真实车辆数据持续改进

核心洞察

平台所有权能满足SDV成功所需的快速创新周期。
掌控了技术根基，您也就掌控了创新节奏。

4

平台与代码库管控 驱动创新

平台与代码库管控 驱动创新

掌握平台与代码库不仅能够解决当前问题，还能开启未来机遇。下面我们将三个关键领域探讨架构掌控能力带来的突破性力量。



以架构控制驾驭AI与ADAS

当前汽车行业聚焦于自动驾驶，AI正在全面重塑车载软件，使其复杂性倍增。对于开发团队而言，这会在安全与合规领域形成危险盲区，并给基于严格安全法规的汽车行业带来严峻的质量保证挑战。

例如，AI和ADAS功能高度依赖于GPU加速代码，但传统的静态分析工具无法解析此类代码，导致现有质量保证工具无法感知的代码区域根本无法实现功能安全认证。

请注意：**Gartner**预计，到2027年，将有17%的网络攻击涉及生成式AI。

若缺乏平台掌控力和完整的代码可见性，您该如何抵御这些新兴威胁呢？

深度融合代码分析的架构掌控可以有效应对这些问题。它使您能够：为AI驱动系统提供ISO 26262合规认证，在保持架构完整性的前提下部署ADAS更新，并在不影响稳定性的情况下迭代机器学习模型。

⁵ Gartner预计全球信息安全支出将在2025年增长15% [↗](#)

为复杂系统的ISO 26262合规而困扰？
[我们的指南可为您提供清晰的解决路径。](#)



核心要点

AI是汽车创新的下一个前沿，但需要平台级架构掌控才能安全、有效地实施并保持合规。

掌控Android Automotive体验

随着Android Automotive OS成为整车厂的热门选择，如何在标准化平台上确保实施质量并实现品牌差异化，已成为新的挑战。

其根源往往在于供应链所采用的“黑盒”模式。为了保护宝贵的知识产权，供应商往往会使用编译后的二进制文件交付应用程序，而非提供源代码。尽管这是行业标准且合理的做法，但给最终责任方整车厂留下了难题，整车厂必须确保数十个集成系统能够提供一致、高质量的用户体验。

平台所有权通过以下能力为解决集成难题提供工具：

- 无需访问源代码即可验证行为的全面测试框架。⁶
- 在所有供应商组件间保持一致的用户体验。
- 在标准化平台上实现真正的品牌差异化。
- 无需重建整个系统即可更换组件的灵活架构。

核心要点

平台所有权可提供相关工具，提高质量与品牌一致性，将标准化的Android Automotive OS转化为独特的竞争优势。

构建数据驱动的用户体验

在SDV时代，核心竞争力是能够基于真实世界的真实数据持续优化的卓越用户体验。由OTA与平台控制所驱动的数据闭环，正是未来营收与品牌忠诚度的核心引擎。

预计到2032年，OTA市场规模将突破130亿美元，但只有具备架构掌控能力、有效管理数据的整车厂才能真正把握这一机遇。⁷

依托坚实的平台基础，车辆系统可以持续学习进化，从而实现：

- 部署**性能优化**，提升车辆响应感知
- 根据用户对车载交互系统的实际使用数据，指导**未来功能开发**
- 启用**预测性维护**等新服务，全面提升用户持有体验

核心要点

架构掌控是开启持续改进循环的关键，它能帮助您利用真实世界数据持续优化用户体验，并开拓新的OTA营收渠道。

⁶ Qt: 《如何简化并扩展Android Automotive的UI/GUI测试》 ↗

⁷ GM Insights: 《汽车空中下载更新的市场规模》 ↗

客户成功案例

 Elektrobit Axivion

客户成功案例：Elektrobit

速度与质量的双重需求

Elektrobit的EB street director导航软件为**奥迪、保时捷、梅赛德斯-奔驰**和**大众**等品牌的车辆提供支持。这些高端制造商对卓越性能和快速交付功能有着极高需求。

Elektrobit在全球各地共有100多位开发者，他们在同一个模块化平台上开展工作。Elektrobit所面临的经典难题是：如何在快速交付新功能的同时，保持架构的完整性。一方面，新加入的开发者需要数周才能理清系统，另一方面每个新增功能还有可能破坏现有功能。

通过持续架构验证解决问题

Elektrobit实施了一套独特的自动化架构验证解决方案，可以持续检查代码是否符合预期基于UML的设计。一旦出现架构违规，即刻就会被发现，而不是拖到数月后的集成测试中才暴露问题。

开发过程变得清晰可见，这彻底改变了团队的工作方式：团队得以参考相关信息，权衡速度和可维护性；开发者可以清楚了解自己的代码对整体系统的影响；新成员可以借助工具而非文档更快上手工作。

客户成功案例

成果：有序架构驱动开发提速

转型带来了显著成效：

- 功能交付更快，变更越界时团队即时知晓
- 新开发者生产效率提前数周
- 基于实际架构影响的项目评估更精准
- 模块化架构在持续演进中仍保持高可维护性

Elektrobit的实践证明，架构规范可以加速开发。清晰边界提升了可测试性，而持续验证在保持汽车安全标准的同时防止技术债务累积。

核心要点

Elektrobit验证了本指南的核心原则：掌控架构即掌控创新节奏。对于面临分布式团队和遗留系统的整车厂，持续架构验证可以将复杂性转化为可管理结构，使质量保证自动化、创新实践可重复。

[阅读完整成功案例](#) ➤



5

前行之路： 专家解读转型路线图

前行之路： 专家解读转型路线图

成功转型无需摒弃既有投资。平台转型可以循序渐进地实施，并在每个阶段持续创造价值。

尽管具体阶段需根据企业目标定制，但成功转型通常遵循已验证的高层模式。为了勾勒此路线图，我们邀请**技术战略总监罗淼**分享了四个关键步骤的洞见。

第4步

扩展成果并建立沟通机制

第3步

培养能力与主人翁文化

第2步

构建原型组织

第1步

获得长期领导层承诺

第1步

获得长期领导层承诺

转型需要坚定的领导层承诺：制定五年以上的路线图、保障专项预算，并由专项SDV负责人在不干扰日常运营的情况下推动自上而下的变革。

“缺乏长期承诺是转型失败的主因。许多传统整车厂在成本激增的首个困境面前就选择放弃。我认为真正的转型意味着坚守信念——坚信这段艰难旅程是必要的，并且终将引领企业走向卓越的未来。”

第2步

构建原型组织

获得领导层支持后，下一步就是在不改变核心生产流程的前提下验证SDV概念。

“组建一支小型跨部门专项创新团队，为现有车型加装新的SDV功能。从软件在线升级(SOTA)开始，逐步过渡到固件在线升级(FOTA)。目标是通过低风险的商业试点项目，在真实场景中验证概念可行性。”

第3步

培养能力与主人翁文化

成功转型需要赋能现有人才，并培育新的思维模式。

“对团队进行敏捷软件开发培训，统一工具链标准，并培育SDV思维模式。根据试点项目反馈持续迭代，既能利用现有人才，又逐渐积累专业能力。最重要的是，赋予开发团队自主决策与执行权。主人翁文化始于个体践行，而非内部宣传中的标语口号。”

第4步

扩展成果并建立沟通机制

模式验证成功后，最后一步就是在保持利益相关者信心的前提下推动整体商业模式转型。

“将经过验证的SDV功能扩展至全系车型，并整合新流程至主要生产线。其中的关键在于向内部团队、外部投资者和客户宣传这些成功案例，以建立并保持转型势头。”



罗森
Qt Group技术战略总监

软件优先转型的投资回报

如前所述，采用软件优先转型的商业价值已获得明确的数据支持。援引Qt Group委托Forrester Consulting在2024年开展的调研，行业领军者分享了采用现代化平台驱动实践后带来的影响。

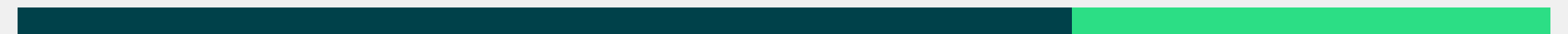
其企业已实现或预期实现的主要收益包括：

如需深入了解该概念，
请参阅《大型企业如何向小型团队学习平台工程》 [↗](#)

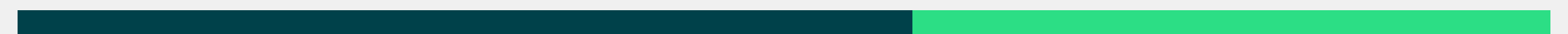
50% 的受访者表示新功能上线时间缩短



68% 的受访者表示终端客户体验改善



57% 的受访者认为嵌入式软件的品牌辨识度与一致性获得提升



49% 的受访者指出缺陷率降低，其中机动车和汽车行业的反响尤为突出



这些显著成果源于一个简单原则：平台驱动战略能够让大型企业在保持企业级规模优势的同时，获得小型团队的敏捷性。

结论

结论

抉择在您手中

软件已彻底重写汽车行业的规则。核心竞争力正源于系统化交付卓越软件的能力，而这一能力建立在架构控制与持续质量保障的双重基石之上。

在本指南中，我们始终将“一亿行代码技术债务”视为创新的首要障碍。平台所有权和代码库掌控能力可作为战略解决方案，并提供专家路线图指引转型路径。

未来的道路已然明晰，一个紧迫的选择正摆在眼前。

选项一：选择被动应对

继续采用碎片化、高投入的方式应对复杂性问题。坐视软件召回侵蚀利润，安全威胁损害品牌声誉，更敏锐的竞争者凭借您难以构建的功能抢占市场。

选项二：掌控技术栈，主导未来

选择主导移动出行的未来。将车辆从一次性交付产品转变为在全生命周期为客户持续创造价值的平台。推动商业模式从一次性交易转变为持续的价值驱动关系。成为定义创新节奏的技术领导者，而不是被动追随者。

成功转型为软件公司的整车厂，将主导未来十年移动出行的格局；而犹豫不决者终将向果断行动的竞争对手寻求技术授权。