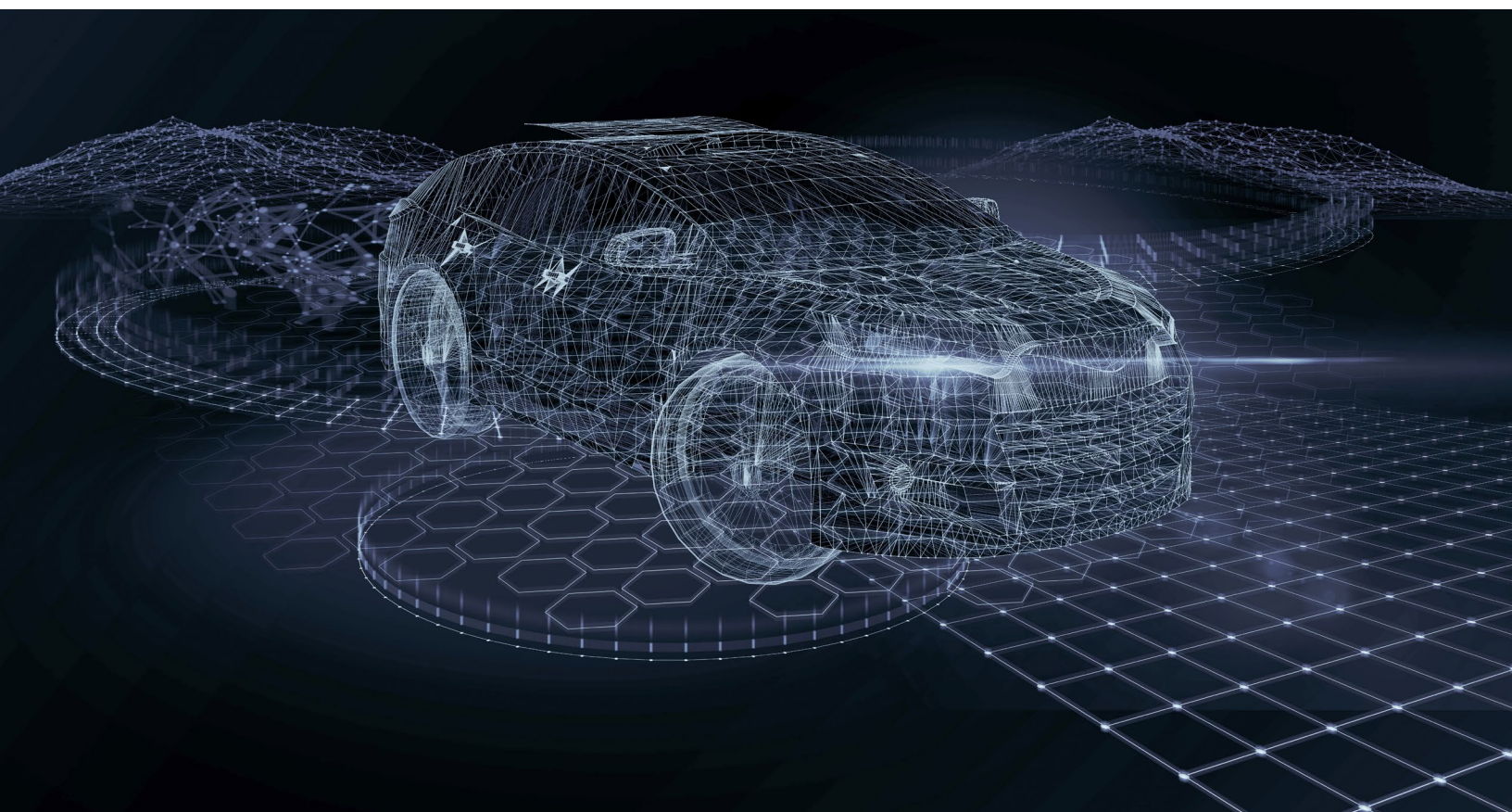


麦肯锡未来出行研究中心

代码为王的时代：车企如何掌握卓越软件开发能力

由于软件正在推动汽车行业的创新，车企研发部门必须迅速掌握软件开发这项复杂能力。

文 / Ondrej Burkacky, Johannes Deichmann, Stefan Frank, Dominik Hepp, and André Rocha



软件正在迅速重塑汽车行业。近年来，业内的四股颠覆浪潮——自动驾驶化、网联化、电动化、共享化 (ACES) ——全都重度依赖先进的软件。这些领域未来还将迎来更多的颠覆发展。全行业的整车厂 (OEM)、供应商和新企业无不希望自己能在这一由软件驱动的新价值链上把握住关键的控制点。

软件: 关键的行业转折点

随着行业格局改变，缺少软件能力的车企将面临重大风险，包括量产 (SOP) 延期和预算超支。它们甚至可能落后于竞争对手和新入场者，倘若后两者能以快得多的速度将新颖得多的产品推向市场的话。更糟的是，软件问题可能导致大规模召回，或是让车企无力防范因黑客攻击而产生的客户安全风险。

我们的研究显示，软件能力强的企业和能力弱的企业之间的差距是显著的，能力最强的企业报告中公布的产量和质量，比能力垫底的企业高出三到六倍¹。这一开发效率差距远大于不同能力的硬件生产商之间可产生的差距。很多车企已认识到强大的软件开

发能力带来的各种优势，也正在采取大刀阔斧的措施改善业绩。一些车企计划在今后几年提升软件能力，并将招聘数以千计的软件工程师；另一些则将重新定义治理模式，建立合作关系，并在全球推广卓越软件开发中心。

然而，我们认为这些措施是不够的，因为只有当车企针对软件开发更新了基础运营模式的时候，真正的变革才会到来。根据我们的研究，在那些认为软件是主要颠覆因素的车企研发领导当中，仅有40%的人觉得自己已为必要的运营变革做好了准备²。虽然一些领军车企已在软件工程实践上取得长足进步，但大部分车企仍然远远落后于佼佼者。目前车企的问题集中在几个领域：敏捷实践、持续集成、自动化测试。

软件开发转型的风险如此之大，车企必须对一整套软件开发方法进行重新思考，包括基础运营模式等。本文分享了我们通过车企、供应商，以及生态圈中其他合作伙伴密切合作，收获的关键理念和洞见。这些洞见还建立在两项基础上：一是针对技术专家开展的广泛访谈，二是利用麦肯锡 SottCoster 专有数据库进行的大规模对标。

根据我们的研究，在那些认为软件是主要颠覆因素的车企研发领导当中，仅有40%的人觉得自己已为必要的运营变革做好了准备

¹ 基于麦肯锡的SoftCoster专有数据库的信息，该数据库包含多个行业的14000多个软件开发项目。

² 麦肯锡未来研发调研。

用数字说话：软件的重要性是如何后来居上的

若干趋势凸显了汽车软件与日俱增的重要性。第一个趋势与软件和电气/电子市场的迅速扩张有关，2020年到2030年，这个市场预计将实现12%的年复合增长率——比普通汽车销量的预期增速高出三倍多。有几个领域增长最为强劲，包括软件功能（年复合增长率达11%）以及集成测试（年复合增长率为12%）³。

复杂度不断升高，但开发效率进步缓慢

无论功能层面还是架构层面，汽车软件的复杂度都在升高，而开发工作的效率却没有以同等速率跟上。我们的研究显示，软件复杂度在过去十年已增加到原来的4倍，而软件开发效率只提升了1到1.5倍（图1）。这个问题在变得日益复杂的大型模块中最为严重，如信息娱乐系统和高级驾驶辅助系统

（ADAS）。相比传统的深度嵌入式软件，开发这些模块的效率大约低25%到35%。

这个日渐扩大的差距可能会影响车企未来的成功。车企需要投入更多资源开发软件，并在软件生命周期中对其进行维护，这可能会削弱自身在创新和应对竞争对手方面的能力。访谈期间，有位企业领导注意到，如果复杂度持续升高，而效率原地踏步，那么仅软件维护这一项，就会迅速耗尽所有的软件开发资源，不会给创新留下丝毫余裕。最终，复杂度和效率之间的差距将削弱成本竞争力，造成严重的财务和商誉问题。

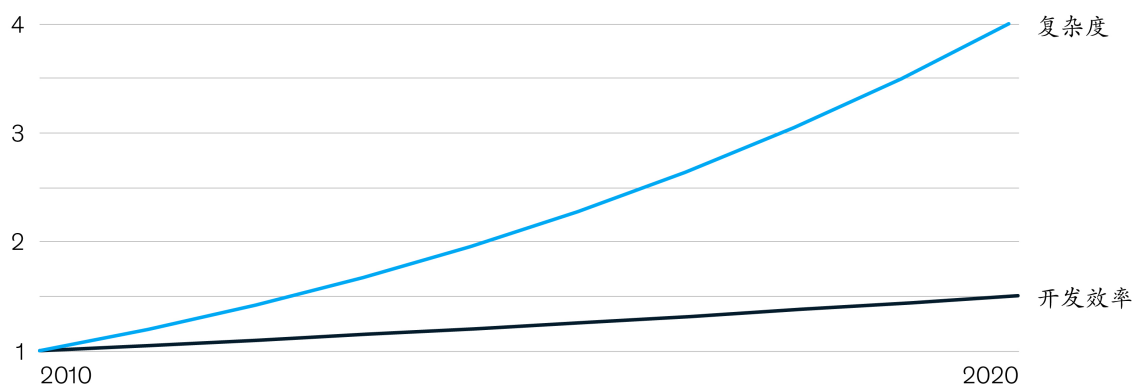
显而易见，软件开发实力排在前25%的企业，相比实力垫底者，效率高出3倍，产量高出3.5倍，质量好上6倍（图2）。因此，它们的产品面市时间更短，软件功能每提升一个档次所需的开发成本也更低。

³ Ondrej Burkacky, Johannes Deichmann, Jan Paul Stein 合撰的文章《2030年的汽车软件与电子市场：梳理行业的未来格局》（*Automotive software and electronics 2030: Mapping the sector's future landscape*）2019年7月9日，McKinsey.com；文章续篇即将出炉。

图 1

软件复杂度提高的速度比开发效率提高的速度更快

各时期软件复杂度与开发效率相对增幅，已根据汽车软件特征做了指数化



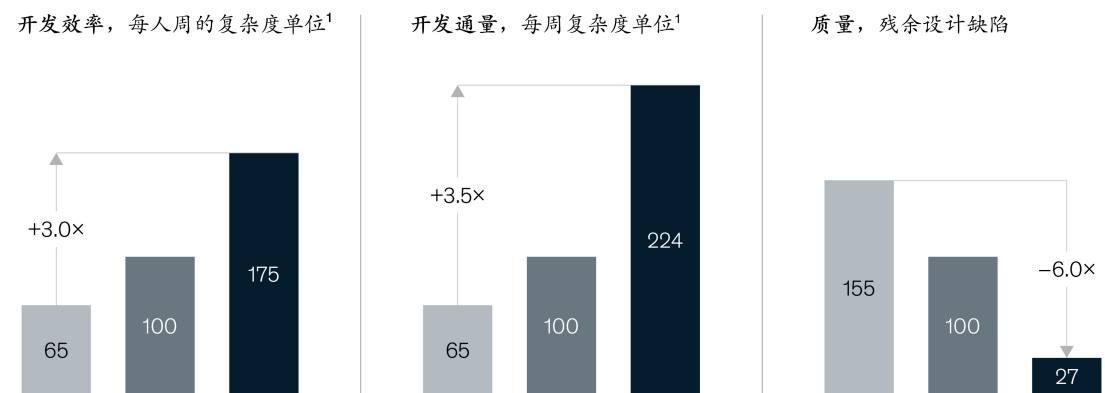
资料来源：麦肯锡SoftCoster嵌入式软件项目数据库

图 2

软件开发实力位居前25%的企业表现出明显较高的开发效率、开发通量和质量

■ 第四分位 ■ 平均水平¹ ■ 第一分位

研发实力处在平均水平和第四分位上的研发组织在研发业绩上的潜在增长



1. 平均水平已指数化，基准值100。每人周复杂度单位代表每个全职员工的开发效率。每周复杂度单位代表整个组织的开发效率。
资料来源：麦肯锡SoftCoster嵌入式软件项目数据库

硬件方面，实力最强和最弱的企业在业绩上的差距相对不那么明显，因此，在这个领域追求差异化的机会也相应更少。

降低复杂度，同时提升效率

为了在这个迅速变化的环境中取得成功，企业应当通过减少开发和维护软件需要做的工作，以求最大限度地降低复杂度。这个策略涉及限制各平台和生命周期阶段的功能和特性的版本数量，同时，企业必须加大对构件的重复利用。至于开发效率，企业应当尝试在软件开发速度上向数字原生企业看齐，以提升开发效率。由于软件创新的整体水平不会下降，企业还必须提高其软件开发和维护的产出量，以便交付在市场上取得成功所需的产品和服务。

降低复杂度和提高效率需要一个新的软件运营模式，该模式着重考量四个维度：

- 开发什么软件，包括架构、设计以及各项要求
- 在哪里开发软件，由企业内部哪个部门开发，包括涉及到的地点、人才及合作关系
- 采取什么方式开发软件，这个维度考量的是开发工作的方法论，如大规模敏捷开发，或是开发和测试流程的改变
- 采取什么方式推动软件开发，包括绩效管理和工具链基础架构

第一个维度——开发什么软件——聚焦的是通过模块化和解耦的硬件/软件架构、以用户为中心的设计, 以及需求管理, 来降低开发复杂度。其他三个维度所聚焦的, 则是通过提供合适的结构、流程和基础设施提高软件开发的效率。我们从四个维度出发, 明确了11项最佳实践, 这些做法能帮助车企成功地解决在软件上面面临的挑战(图3)。理想情况下, 企业将同时处理好所有维度。

A. 开发什么软件: 架构、设计及各项要求

在新的运营模式下, 企业必须把与软件相关的开发目标和业务机会转化为产品、功能和模块等层面上的可行架构、产品及投资组合需求。通过这个流程, 企业可以详细了解能为其创造价值的软件类型。这个流程还将助力企业降低架构的复杂度, 应用以用户为中心的设计流程, 改善对软件需求的管理。

图 3

新的软件开发运营模式需要对四个关键维度做出改变

各维度的最佳实践

应用以用户为中心的设计	对软件需求的管理方式进行调整	调整组织并建立全球卓越中心	确保企业有能力获取顶尖软件开发人才资源, 并对其保持吸引力
降低架构的复杂度	A 开发什么软件	B 在哪里开发软件	制定清晰的“自制或外购”策略并建立合作生态圈
	D 如何促进软件开发	C 如何开发软件	实施大规模敏捷方法 实现硬件与软件开发之间的解耦
升级到标准化的先进软件开发工具链		提高测试的自动化程度, 实现成熟的持续集成	

A1. 降低架构的复杂度

根据我们的研究,如果汽车软件的模块化程度不够,就会使设计复杂度提高,也会提高项目的整体难度。另外,汽车软件的架构构件边界往往是不理想的,这有可能导致更多的相互依存关系,使开发人员在添加新功能时必须修改的构件数量成倍增加。这些依存关系还可能造成一个后果:检测到缺陷时,需要更长的时间和更高的专业水平来追踪特定软件模块和开发团队的错误。

为解决此类问题,OEM必须大幅提高标准化和模块化程度,这样就能实现软件在各平台之间的可扩展性,使软件复杂度维持在可以管理的水平。车企还必须重视两项工

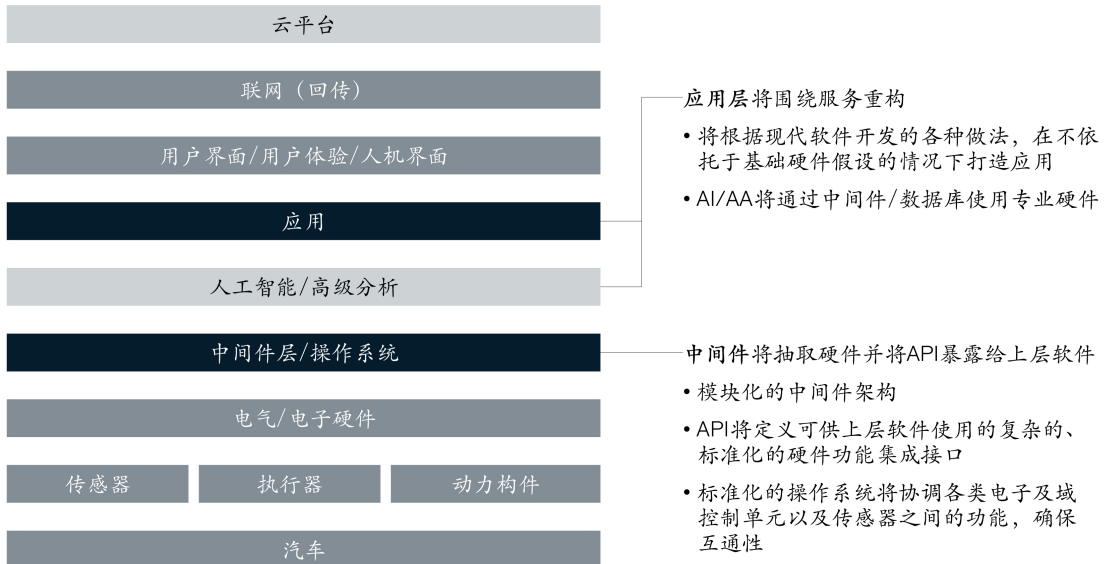
作,一是促成软硬件之间的解耦,二是应用服务导向型设计。

解耦架构。车企可以引入一个强大的中间件层次,利用它提取硬件能力,并通过上层使用的标准化应用程序接口(API)使这些能力对软件功能开放(图4)。这种软件架构可以利用各平台之间的共性,降低设计上的复杂度,不必多次重复开发相同的软件。

除了解耦,车企还应当创建一系列标准化的操作系统,利用它们支持各构件之间的协调,确保互通性。很多企业已宣布将要开发此类操作系统,但目前尚不存在统一的方法。

图 4

车企必须力争实现一个理想架构,使之能支持软硬件之间的解耦,而且具备强大的中间件



而且企业也还没有明确这些系统的确切研发重点和功能。

通过遵循明确的架构原理和指导原则,企业能够有效地应对更高的系统和软件复杂度。硬件/软件解耦允许多个实体参与模块化开发。反过来,由于代码共性增多,模块化软件构建技术可增加对代码的重复利用,并减少需要的代码总量。很多企业已着手引入软件产品线工程(PLE)方法,以增加重复利用,同时处理产品变更。这个方法能让一个软件服务于多个产品、产品变体和产品系列,还可以兼容不同的硬件版本。软件PLE显著降低了开发和测试工作的难度,这两项工作都只需进行一次。

换句话说,促使硬件从软件上解耦,可促使硬件实现进一步的“自主化”,硬件提供了标准的计算、存储、输入/输出和电源功能,而软件则定义了终端用户功能。对于需要标准性能的应用,不同的软件功能可以利用虚拟化和容器化技术在相同的硬件上运行,如有必要,还可以动态地分布到其他硬件(例如在硬件发生故障的情况下)。对于ADAS等存在实时性能要求的应用,针对特定硬件的软件开发工作对于实现最优效率仍然至关重要。

以服务为导向的架构。该架构应当遵循各项服务的定义,反过来,这些定义则应将业务或用户需求代码化。以服务为导向的架构设计既能让企业实现各核心要素的标准化,也能让各部门和各业务单位之间的接口实现标准化。企业还应对单项硬件和软件要素应用标准化的设计,以便在不影响性能的情况下,针对其他支持设备和功能规模化地配置资源,如计算和存储资源。以服务为导向的架构对OEM尤为重要。实现从车辆

到云端的迅速连接,不仅将提升车企各种车型的长期价值,也将推动车企能力的迅速更新和升级。

A2.应用以用户为中心的设计技术

综观各行业,专注于开发以用户为中心的强大设计、创造最优用户体验(UX)的企业,将实现更高的财务收益⁴。随着ACES概念持续受到追捧,软件定义的汽车成为常态,对于OEM的整体竞争力而言,这些特征将会变得越来越重要。即便是顶尖企业也需要做改进,原因在于汽车行业在设计良好的软件用户体验、提供最优的客户价值等方面仍然落后于其他行业。

按照最佳实践奉行的设计原则,OEM应当与终端用户合作对新软件进行迭代,无论交付前后都应如此。车企也应当采用新的交付模式,以便以每周或每月一次的频率对软件进行更新或添加,从而实现持续的优化。这些工作的周期相比经典软件开发工作短了许多,后者通常需要几年时间。新的交付模式还有一个好处,它可以让OEM与客户实现持续的直接接触,从而使OEM持续收到反馈,而这些反馈有助于它们优化软件需求,提供积极的用户体验提升。当OEM依赖硬件实现升级时,此类互动是不可能实现的,因为接触只会在通过经销商网络进行的一次性销售或市场调研期间发生一次。为了达到这个目标状态,OEM需要落实必要的先决条件,例如配套的软件和电子架构,以及可实现云端更新的工具链。

希望成为用户体验领导者的OEM必须对自己的数据善加利用。随着车载软件和传感器数量不断增多,OEM如今能够获取大量的数据,以了解客户如何使用汽车。OEM可以挖掘此类数据,识别对客户最重要的

⁴ 基于针对业务发展的麦肯锡设计指数评分的相关分析。如欲了解更多信息,参阅Benedict Sheppard, Hugo Sarrazin, Garen Kouyoumjian和Fabrizio Dore合撰的文章《设计的商业价值》(The business value of design), McKinsey Quarterly, 2018年10月25日, McKinsey.com。

成功的软件开发需要根据客户反馈对需求进行持续的调整和修正

特性, 以及配置超标或根本没人用的特性。这些洞见将为明确配置和未来型号需求提供参考。

最终, 新的交付模式将给开发效率带来积极影响。鉴于车企将对软件进行频繁的变更、调适、修改, 它们不需要从项目一开始的时候就确定极端详细的需求。由于用在定义需求上的时间缩短了, 产品面市时间也将缩短。

A3.调整软件需求管理

历史上, 汽车行业曾是在集成的价值链上管理各项需求的领导者。但是, OEM主要关注的是硬件需求, 而且它们的既有流程并未针对软件调适到最优状态。随着车载软件成为重要的差异化因素, OEM必须采用新的做法来管理需求。管理需求的变更尤为重要, 因为我们的研究显示, 对汽车软件的种种需求已经变得过于细化, 以至于拖慢了开发进程。

按照最佳实践, OEM应当基于客户价值对各项需求进行聚类。第一个层级应当主要包括面向客户的需求(通常被表述为用例)。另一个层级主要包括技术或实施方面的需求(通常被表述为赋能因素), 比如某个特性所需的内存。这个方法可以保证OEM着

重关注价值创造, 并能在软件开发期间设定合理的优先事项。随着车企将各种需求划分到多个层级, 以下工作能够起到帮助作用。

将需求与战略和客户价值挂钩。成功的软件开发需要根据客户反馈对需求进行持续的调整和修正。虽然企业在初期就应根据其商业战略和目标定义软件需求, 但它们也应根据客户反馈和开发进度周期性地做出调整。

确保端到端的可追溯性。通过在整条价值链上对需求进行密切追踪, 车企可避免做无用功, 加快开发进程。但是, 只有当车企的开发流程和工具链能够从定义到验收全过程实现需求的严格可追溯性时, 车企才能做到这一点。这种明确性有助于车企了解清楚各项需求(客户观点)、需要的功能(开发者观点), 以及各项交付成果(测试者观点)。

车企必须分四步实现端到端的追踪, 这要么需要用到少数几种高度集成的工具, 要么用到四种专业工具, 并搭配相应的接口。这些步骤是:

- 对需求进行追踪, 从特性到构件对这些需求进行细化和具体说明

- 管理待办清单，这项工作有助于各团队在软件开发冲刺时管理好各项需求的覆盖范围（与下一个步骤密切相关）
- 追踪代码变更，包括对代办项目的更新
- 通过开展测试案例对需求进行验证，并检查测试案例的通过-失败状态

利用工具将各项需求关联起来，用户能够高效地在项目的每个阶段实施变更，从而满足监管对端到端可追溯性的各种要求（例如，ASPICE或UNECE⁵）。利用这个方法，我们能迅速弄清楚哪些变更影响到了哪些工作成果。当遵循敏捷流程开发软件时，此类需求变更是正常的，也是可取的（如基于客户反馈的变更），企业应当利用各种流程和工具对它们予以支持。在软件开发工作的传统瀑布式流程当中，此类变更是罕见的，而且通常也预见不到。

避免过度细化，创建明确的需求类别。企业可以建立一些最佳实践，对软件需求进行具体说明和分类，并拟定精简的测试办法。一份优秀的需求说明应当是清晰明确的，而且允许测试工作不受其他需求影响。与组合管理一样，企业应当对不同类型的需求加以区分。常见的需求类别包括法律监管类、安全类、战略类，以及必要改进、客户价值、成本赋能等因素。另外，企业还必须确保，凡是需求之间的依存关系都应当是透明的。很多企业已经将这些规则融入到自己的软件开发流程和培训课程之中，以便优化对需求的处理和评审过程。

开展优先排序，进行持续调整。企业既应根据具体的商业论证和战略目标，也应根据在整个开发过程中（例如在测试过程中）获

得的客户反馈和和经验对软件需求进行评估和优先排序。企业应当定期对需求进行重新评估，并在一份公开透明的待办清单中对其加以维护。

很多企业任命了产品负责人，这些人拥有宽广的知识面，可以做出产品功能取舍，组建跨职能团队，并确保各职能部门就各项需求达成共识。根据最佳实践，产品负责人还需要负责遵循最佳实践，并对需求和用例的待办清单进行维护。

B. 在哪里开发软件：组织、地点、人才与合作伙伴

大部分车企缺乏处理大规模软件开发工作所需的组织基础。它们面临多重挑战，有的在执行层面极少或没有划分出明确的职责，有的没有足够的软件工程师和架构师。凭借明确软件开发工作所需的组织架构、地点和人才策略、“自研或外包”策略，以及所需的合作关系生态圈，新的运营模式将会解决这些问题。

B1. 调整组织，建立全球卓越软件中心

在组织层面，大部分车企还没有做好准备，无力应对致使软件重要性攀升的ACES趋势。例如，OEM就软件问题进行决策时通常比较缓慢，很多车企对于车载平台的软件、电子设备策略，以及相关预算的主要权责尚无明确的划分。为了在新形势下保持竞争力，车企必须重新思考其组织设置。此项工作的一个主要目标，是要在端到端开发流程期间减少在架构定义、需求定义以及开发这几项工作之中的接口。通过增进各团队在各阶段的共识和协作，车企可以避免做多余的工作，并优化各种既有的和全新的能力。

⁵ 汽车软件流程改进及能力评定（Automotive Software Performance Improvement and Capability determination）；联合国欧洲经济委员会（United Nations Economic Commission for Europe）。

很多车企已成立一个集中化的职能部门，依靠该部门负责软件架构设计，并分享已固化的最佳实践。例如，一家OEM最近成立了一个中央职能部门，该部门聚集了5000名软件开发人员。然而，大多数情况下，OEM对软件开发仍采取较为去中心化的方法。

*几种组织类型。*尝试改进软件开发工作时，OEM可以采用几种常见的组织类型。对于每家企业而言，能反映公司优先事项的选项，就是最佳选项，这些优先事项包括加快决策、减少接口、明确职责等。更重要的是，企业必须开展专项工作，以便协调来自不同组织职能的要求，这些职能包括研发、采购、生产、销售和后市场等部门，这是因为这些职能对软件的需求和生命周期变更可能彼此存在冲突。此举将确保无缝的用户接口和高效的开发流程。除了重塑组织，车企还必须对弥合软硬件团队文化差距和协调工作流程进行大量投入以开展相应行动。这些工作需要持续的变革管理，但它们将有助于车企成为软件开发重镇。

在定义组织架构时，汽车软件开发单位将对职能角色、产品或项目，以及技术构建理想图景。不过，这个组织架构一般会从前述的几个维度中选择一个作为核心。

在此可参考一个强调职能角色的组织架构。在这种模式下，企业将从软件职能组织抽调个体成员，派往针对特定产品或平台的项目。产品和技术将通过面向各职能单位的间接、“虚线式”的汇报架构得到体现。这种架构为企业赋予了最大的灵活性，因为它们不需要为了适应产品/项目和技术方面的额外要求而重塑组织。然而，这种架构之下的开发效率可能不是最佳的，这会使成立跨部门职能变得更有必要，如项目管理、架构和人员配置部门。

在第二种类型中，组织架构侧重的是项目，诸如针对特定客户、车系、车型乃至平台的项目。这种类型可利用“虚线式”汇报条线和成熟流程将产品和技术这两个维度联系起来。这种类型使得客户和终端产品成为组织的首要焦点。从不利方面来看，它提高

尽管车企必须在诸多方面表现优异才能赢得这场软件竞赛，吸引和留住顶尖人才很可能才是最重要的维度

了冗余工作的风险,也在不同的产品或项目团队之间造成了各种障碍,这可能对技术移交形成干扰。强大的平台和架构有助于缓解这些风险。

在第三种类型中,组织架构侧重的是技术和专业领域,如网络、人机界面,或是后端。在这种模式下,企业从技术部门抽调个体成员派往具体产品的项目。通过虚线汇报和成熟的工作流程,这个方法能实现对技术和职能这两个维度的聚焦。虽然这个模式有助于养成深厚的技术和专长领域,但它在项目范围、需求、规格方面的灵活性很差,即使这几方面在项目期间发生变化时也是如此。在开发和维护多种产品时,企业通常会采用这个类型。

B2. 获得并吸引顶尖人才,开展内部能力建设

虽然车企必须在诸多方面表现优异才能赢得这场软件竞赛,但吸引和留住顶尖人才很可能才是最重要的维度。

大部分OEM已将软件开发工作大量外包并依赖战略合作伙伴关系。ACES趋势极大地提高了软件的重要性,即便软件开发效率有迎来几番增长的潜力,2030年前对于软件工程师的需求仍可能增加三到四倍。由于汽车行业正在与科技企业以及其他行业正面竞争软件人才,车企需要采取较为大刀阔斧的举措才能改善顶级开发人员的招募工作。否则,不断扩大的人才缺口还将继续扩大。

汽车软件人才的招聘和挽留计划应当侧重于实现差异化。根据我们的对标分析结果,

相比只具备同质化经验的团队,具备多样经验的团队在开发效率方面的绩效要高出两位数的百分比。企业可以通过开展以下几项工作改善人才招募现状,

要加大对全球软件开发人才资源的利用。综合选址策略有助于企业扩展其软件开发活动、构建相关能力并提升产能,同时确保成本可控。而这也有助于企业争夺顶级人才。一些创新企业已经在数字人才热点地区建立了新的汽车工程中心,例如自动驾驶中心。不过,大多数传统型企业在很大程度上还是保留了其历史足迹和硬件中心,但这可能使其吸引和留住软件人才的努力变得复杂化。如果这些企业在关键区域的多个地点开展业务的话,就可以从附近的大学和其他教育机构吸纳人才。为加强联系,这些企业可以与大学合作开展访学实习计划,使其能够接触到应届毕业生和其他专业技能人才。

尽管全球足迹可带来诸多好处,但也需要合适的运营模式,才能避免远程和/或分布式工作的普遍问题。例如,研究表明,开发项目每增加一个新站点,软件开发效率平均会下降约10%。

使企业对软件人才更具吸引力。我们的研究表明,薪酬、职业发展机会、工作类型和企业文化是留住软件工程师的首要因素。目前,车企在上述这些方面均落后于科技企业。为弥合差距,前者必须针对所有重要的影响因素制定独特且有针对性的雇主价值主张,不能再简单地固守传统福利,如就业保障和企业配车等。

由于汽车行业正在与科技企业以及其他行业正面竞争软件人才，车企需要采取较为大刀阔斧的举措才能改善顶级开发人员的招募工作。否则，不断扩大的人才缺口还将继续扩大

鼓励内部能力建设。一个合理的做法是，尽可能重新培训目前以硬件为主的员工，让其在技能升级后承担软件相关职责。例如，企业可以培训硬件项目经理来监督软件项目。但是，只有当企业在软件专业人员和经过再培训的硬件专家之间保持良好的平衡时，这种做法才行之有效。这个平衡视具体企业情况可能有所不同，而且难以量化。不过，根据经验，许多企业将比例保持在2:1（即2位软件专业人员:1位再培训硬件专家）时，效果最好。要获得最佳结果，企业应制定有吸引力的发展计划和在职培训计划，帮助员工快速学习新的软件语言和方法。此外，还应强调终身学习。

尽管再培训可能会弥补许多人才缺口，但企业不应忘记，大多数人都是在工作多年后才发展出根深蒂固的解决问题模式。出现意外问题时，硬件背景的软件员工可能采用固化的问题解决方法，而这些方法可能更偏线性而非敏捷性。因此，企业应尝试在培训期间鼓励采用新的思维方式。若能成功的话，软件专家和再培训员工之间的差异将迅速缩小。相反，忽视这种做法则可能妨碍整个组织的敏捷转型。

提供职业路径。与技术企业相比，大多数车企在明确职业路径和成长机会方面仍存在不足。之所以出现这种差距，是因为专家职业路径在车企还未广泛普及。此外，OEM也很少定义职位期望和资历级别。想要继续晋升的专家就只能担任管理职务，因为没有其他选择。

为了提高留存率，车企可以引入明确的职业路径，每个级别都有特定的技能要求。一些路径面向业务专家，另一些则针对管理人才。晋升路径可以是纵向的（从初级到高级开发人员再到团队负责人），或者同样重要的横向轮岗，持续六个月或以上，侧重培养不同的技能，包括与领导力、项目管理和软件开发有关的技能。企业还应制定专门的培训计划，包括职位和跨学科培训，开放给更大范围的员工。明确的职业路径有望提高效率，因为专家通常比新手效率更高。

B3. 明确清晰的“自研或外包”策略并建立合作生态圈

要保持来之不易的竞争优势并避免成为同质化的硬件平台开发商，车企必须制定明确的客户战略，确定差异化配置和价值主

张, 创建可以对开发模块进行逻辑性分解的模块化体系架构。完成这些任务后, 车企即可着眼于三大维度, 从而制定明确的“自研或外包”策略 (图5):

- 开发工作所在阶段, 例如系统集成或验收测试
- 软件技术栈
- 软件领域或模块, 可能涵盖信息娱乐或动力总成等领域

企业应确定并定义这些维度上的控制点, 从而基于其整体策略 (例如, 针对关键知识产权、质量标准和差异化创新等) 确定“自研或外包”策略。

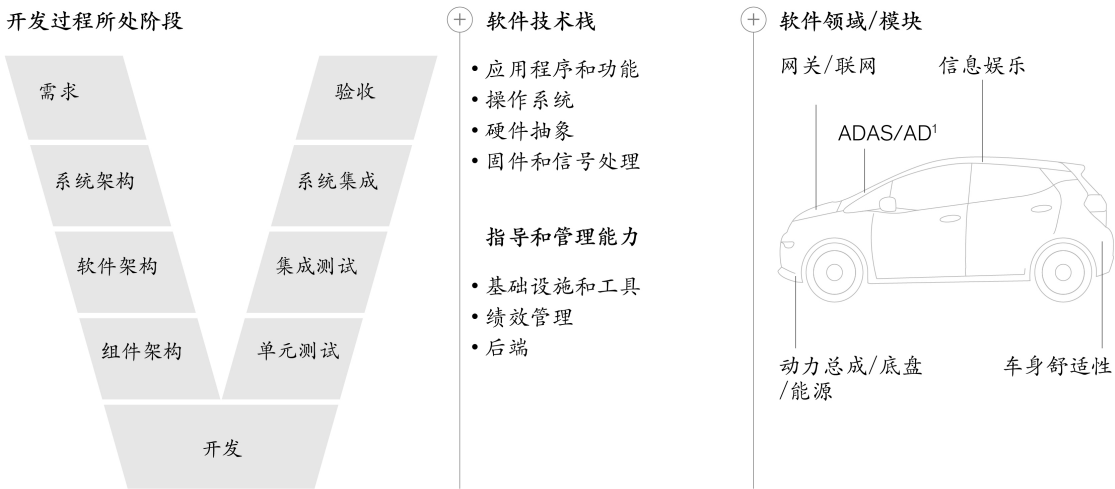
当然, 企业在决策过程中还必须考虑市场采购的难易程度。其他因素 (例如成本) 也很重要, 但通常不是决定性的。要改善决策, OEM可基于优先级和市场情况权衡各个因素。

企业决定内部自行开发软件时, 必须评估对内部工程能力的影响, 确定现有人员是否具备所有必要的的能力, 并检视组织架构和流程。如果企业缺乏合适的能力或能力不足, 则应探索收购或合资机会, 从而确保其对关键控制点的把握。

如果企业决定外购软件, 则必须在扩展评估过程中明确具体的采购模式, 扩展评估涉及开发合作伙伴的筛选和签约工作。针对复杂软件系统考虑部分外购时, 企业最多应签约两到三家供应商。

图 5
企业在进行“自研或外包”决策时应考虑三大维度

汽车示例



1. 先进的驾驶员辅助系统和自动驾驶

传统车企一直更重视硬件，其次才是软件。 现在必须重新审视这种观点以及开发方法， 因为软件是当前产品开发组合中一个主要的 价值驱动力

我们的研究表明，超过这个数量就会导致开发效率下降65%以上。

在全面的“自研或外包”策略中，企业应采用标准的开源组件，因为它们在软件开发过程中可提供巨大的优势。不过，企业需建立明确的开源模块使用规则和流程，并注意许可、责任和维护问题。通常，OEM和供应商需签署正式的法律协议，才能在产品中使用开源组件。

最后，车企应发展战略合作伙伴关系，确定生态圈合作企业，因为这些联系有助于相互学习，同时加快开发速度并保持较低成本。共同开发还可降低较晚进入市场所产生的风险。

在“自研或外包”策略中，OEM会将差异化功能的生产留在内部，而将非关键软件的开发外包给其他供应商或承包商。除前述优势外，该方法还将大大减少对软件人才的需求。

C. 如何开发软件：敏捷、解耦、测试

传统上，车企一直都更重视硬件，其次才是软件。现在必须重新审视这种观点以及开发方法，因为软件现在是产品开发组合中的一个主要的价值驱动力。这个转变要求车企实施大规模的敏捷方法，解耦硬件和软件开发流程，同时提高测试自动化和持续集成的水平。

C1. 实施大规模敏捷方法

敏捷方法在硬件和软件上得到大规模应用时，可使企业提高开发效率并快速应对环境变化。通过敏捷转型，各行业企业都实现了开发效率和实施速度30%的提升，同时将发布时的残留缺陷减少了70%以上。⁶ 因为降低了项目预算、时间和质量方面的风险，所以敏捷在应对复杂性挑战上发挥了至关重要的作用。

迄今为止，只有少数车企自始至终地推广敏捷软件开发的实践做法。尽管许多企业都在试点，尤其是在高级开发项目上，但只有少数真正大规模实施了敏捷方法。

⁶ 麦肯锡的SoftCoster嵌入式软件项目数据库。

采用率低的原因可能在于汽车行业的应用有非常具体的要求,因此很难在整个组织中实施标准的敏捷方法。此外,汽车行业使用的敏捷工具必须能够处理系统复杂性、与硬件开发之间复杂的相依关系、以及网络安全、车辆安全性和质量相关的严格法规要求。

与硬件开发的依存关系。要有效管理相依关系, OEM可将系统工程与需求管理技术相结合,采用大规模的敏捷方法。这些做法可确保流畅的软硬件集成,以及开发时间线的同步。

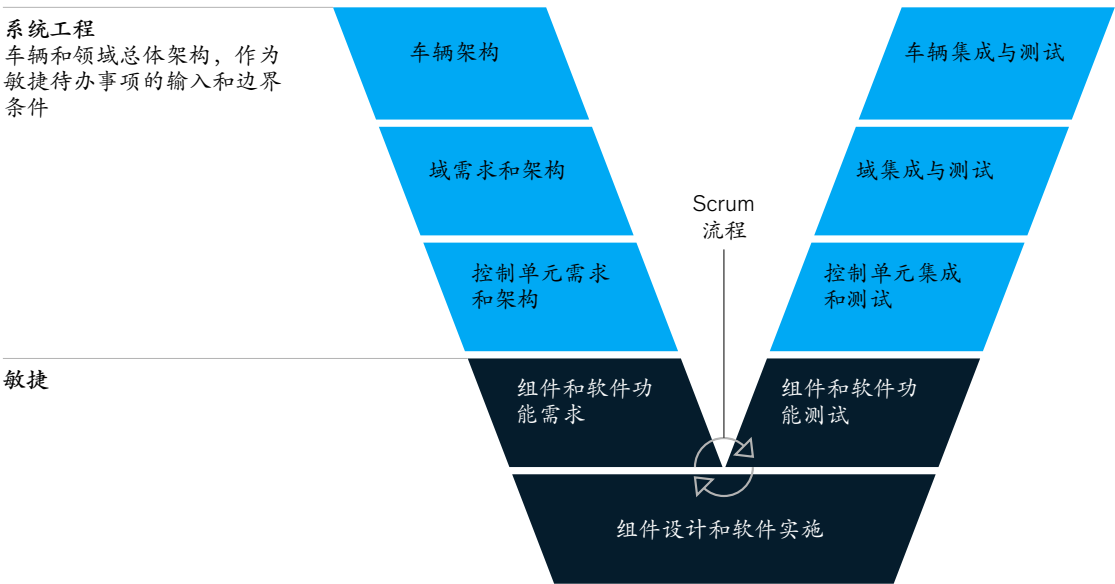
涵盖总体架构、集成和测试的经典系统工程实践做法将确保集成化的产品生命周期管理,并帮助企业满足法规要求。企业可利

用系统工程实践,明确车辆和领域整体架构(图6)。反过来,这又为敏捷团队提供了高层输入和边界条件。基于这些输入,敏捷团队可以先细化软件需求,再进行组件的开发和测试工作。开发周期结束时,当领域、车辆集成和测试活动合成完整系统后,团队即可关闭系统工程回路。

从项目管理的角度来看,目标是使各部门对控制单元和领域架构专用要素的优先级和所需同步点达成一致意见。例如,企业应优先考虑并跟踪某些功能的交付情况,因为这些功能的启动对于时间敏感事件如冬季测试,是必不可少的。这样,在看总体需求完成指标时,就只需监控那些不太重要的功能了。

图 6
车企应将总体系统工程与敏捷软件开发相结合

系统工程与敏捷相结合



法规要求和行业标准。敏捷开发做法在汽车行业环境下完全可行,且大大提升了效率,但企业必须考虑法规以及硬软件的基本同步问题。

因此,可能需要更改部分传统的工作流程,例如创建认证组件的方法(从而符合ISO 26262标准)。在传统瀑布式开发模式下,认证组件的生成顺序与工程活动相同,即规划、客户需求、系统架构、软件需求、软件实施和软件测试。而在敏捷开发模式中,最好的方法却是反向认证,即在项目结束时,也就是所有软件需求和代码都确定、固定之后,创建认证组件。这种做法可最大程度地减少浪费,因为不用多次生成认证组件,并且需要软件安全审核员从项目支出保持一致,且与团队关系良好。软硬件解耦有望进一步简化ISO 26262认证工作,因为重复使用的软件可能会通过经验证的参数进行合规验证,

即当其他应用程序已经使用该软件而未出现问题,则证明可使用该软件。

目前,诸如ASPICE之类的行业标准规定所有需求都必须可追溯,并能够审计所使用的流程和工具。需求的可追溯性与敏捷实践做法兼容,可通过自动化工具链高效实现(请参阅A3部分的需求管理,以及D2部分的标准化工具链)。但是,流程和工具的审计需求可能会限制敏捷技术固有的持续改进系统。尽管“敏捷”团队可独立地改善其流程和工作方法,但汽车团队却必须符合文件标准的要求,并确保各个团队之间的同步,这些行动可能拖慢其进度。

何时采用敏捷方法。尽管需要预定义的待办事项以及可审计的流程和工具,汽车软件团队仍可即刻采取大多数敏捷实践。但是这会极大地改变他们的工作方式。

敏捷开发在汽车行业环境下完全可行,且大大提升了效率。但企业必须考虑法规以及硬软件的基本同步问题

转向敏捷方法时, 组织必须对宏观层面的运营 (包括项目组合、资源和项目管理) 进行关键设计的决策。通常, 只有先进的开发部门, 例如OEM的“软件工厂”, 才采用规模化的敏捷方法, 即所有部门都完全采用新的工作方式。但是也有例外, 例如, 一些汽车先驱企业已实施了规模化敏捷方法, 例如规模化敏捷框架 (SAFe), 指导其总体研发工作。

与之相反, 各个团队则应始终遵循已建立的敏捷实践开展业务。例如, 跨部门代表和/或团队成员同址办公以及有时间限制的迭代都非常重要。与其他行业一样, 在将敏

捷方法应用于负责各个功能的团队时, 其优势可能最为明显 (图7)。

当OEM向敏捷流程过渡时, 还必须:

- 将开发供应商集成到其敏捷流程中, 促进全面的应用
- 调整采购流程, 从有明确预定义和预协商的基于规格的合同关系转变为基于冲刺的开发合作伙伴关系
- 解决影响供应商与OEM共址办公或敏捷合作的法律问题

图 7
敏捷方法可推动多个维度上的变革

变革示例 (非穷尽)

类别	示例	从	到
团队架构	产品团队设置	兼职人员	全职人员
		每个部门都有独立的团队	完全跨部门的团队
工作方式、角色和责任	基于结果的指导/目标	基于活动	基于结果或KPI指标
		比照预先定义的项目计划进行衡量	持续评估
	连续发布的节奏	瀑布式, 每年3-4次	敏捷方法, 每周或每天
	决策	层级很多, 由高层决策	产品负责人 (决定开发什么) 和敏捷团队 (决定如何开发) 负责决策
技术与架构	模块化或解耦化软件架构	单体架构	服务导向
	可扩展的IT基础设施	刚性架构; 有时IT系统和数据模型都不完整	精简和可扩展IT基础设施支持的模块化IT架构

C2. 软硬件解耦，促进双速开发流程

在流程方面，车企可采用更动态的软件周期计划，支持经常性的发布，而非受限于严格而遥远的车辆平台SOP日期（图8）。将产品及生命周期管理与硬件解耦是脱离独立整车（one-vehicle）SOP方向的关键。为此，必须保持独立的待办事项和路线图，同时明确软硬件开发之间的同步里程碑。在与供应商合作时，OEM可能需要重新签订新的协议。最后，OEM必须加强自动化软件的使用，集成测试和部署。

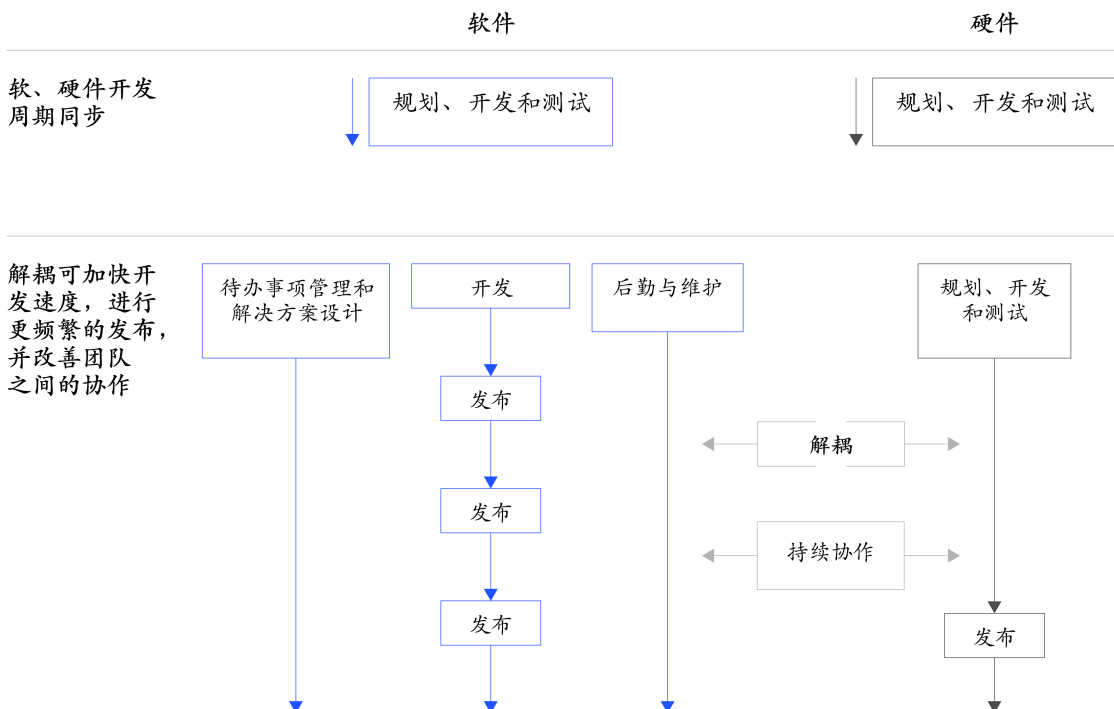
与敏捷方法一样，系统开发团队可管理和定义硬件和软件团队之间的接口，明确划分硬

件/软件待办事项，确保各层级的同步。显然，如果硬件和软件彼此独立，即可分设架构冻结点，这样就无需同步了。

像敏捷方法一样，解耦也需有若干先决条件，包括标准化和模块化的架构以及可靠的测试方法。如前所述，企业可以使用强大的中间件将软件与硬件解耦，中间件可以提取硬件能力（middleware layer that abstracts hardware capabilities），通过标准化API将其提供给需要的功能和服务。但关键的解耦抓手其实是具有鲁棒性的车辆测试方法。因此，企业必须定义提供和维护测试车辆的方法：例如硬件在环或软件在环系统，或者是更广泛的仿真基础架构。

图 8

企业通过解耦软件和硬件开发周期，从而获得许多优势



敏捷方法和硬件/软件开发解耦都对价值链的下游有重大影响,尤其是对于采购组织而言。例如,采购需从传统的瀑布式采购流程转变为更适合敏捷和解耦开发方法的模式。在实践中,这就要求OEM遵循某些最佳实践做法,例如不断评估软件要素对于战略的重要性,了解需求将如何演变以及确定每种情况下最合适的采购模式。这些变化要求对软件总拥有成本以及新的合作模式有所了解,新模式侧重战略合作伙伴关系而非多供应商采购。

C3. 提高测试自动化、完善持续集成,确保尽早发现错误

随着软件数量的增加以及对连续功能更新的更大需求,OEM必须尽快发现并解决漏洞和接口错误。如果无法直接解决问题,漏洞可能会大量积压,从而消耗掉所有资源并使问题跟踪复杂化,导致开发延误并产生更多的验证工作。

与敏捷实践一样,几乎没有车企大规模采用持续集成或自动化测试的方法。但只要它们采用这些做法,即可降低发布风险,同时大大提高开发效率。根据我们的经验,企业可将开发效率提高40%以上,同时将残余缺陷密度降低60%以上。

要仿效先行企业,车企应采用两个相互关联的软件开发最佳实践。首先,应每天几次将代码集成到共享存储库中并通过自动构建进行验证。尽早集成代码有助于开发人员“快速失败”(及早发现问题),然后通过持续集成实践、工具和自动化手段,轻松隔离问题,使之不会产生更大的问题。供应商则可独立地在系统层面上获取这些益处。在整车层面上,OEM需要解决IP限制问题,

自研编码(insource coding)或采用“白盒”方法,与供应商共享完整代码。解决方案的第二个要素是测试驱动开发和自动化流程,即在编码开始前就对测试进行定义,然后在代码集成后自动进行测试。

软件设计人员和开发人员(而不是独立的测试部门)在开发过程中与客户一起对测试进行优化和迭代。该方法迫使开发人员在编码开始前即要考虑如何使用系统以及如何实施这个问题。最终,全面的自动化测试套件将使可持续、冲刺成为可能。

D. 如何促进软件开发:性能和工具链

为了优化软件驱动型运营模式的效率,企业应采用一流的性能管理方法,并通过构建软件开发工具链,建立最先进的软件开发基础架构。

D1. 实施性能管理,涵盖数据驱动的开发效率、项目成熟度和质量度量

随着软件复杂性的提升,车企必须升级其性能管理体系,采用标准化、数据驱动的指标对开发效率、项目成熟度和质量进行度量。只有自动化的、数据驱动洞见才能赋能基于事实的实时性能管理方法,主动暴露出时间、成本和质量方面迫在眉睫的软件问题。

一流的软件性能管理系统应在以下三个关键事项上脱颖而出:

- 建立全面的KPI(如成本、时间和质量KPI指标)体系,确保每个指标都与总体业务目标和运营任务挂钩

引入标准化、最先进的开发工具链，是释放自动化测试和敏捷方法30%至40%开发效率潜力的关键推动力

- 开发一个有效的问题上报系统，包括标准会议这种形式，侧重简单、快速的汇报、决策和上报
- 采用工具化、自动化的开发效率测量技术和代码质量衡量指标，确保对软件开发效率和质量进行客观的测量
- 模块的成熟技术。根据我们的研究和经验，先进的工具链可以：
 - 推动更快、更敏捷的开发，将代码发布或构建所需工作减少90%
 - 通过严格的质量管卡的使用而降低缺陷密度，从而有可能将故障率降低50%
 - 允许企业每天进行成千上万次编译，而无需任何人工干预

D2.升级开发工具链

车企可引入标准化软件开发工具链，支持持续集成和标准API的使用，从而提升开发效率。典型的工具链组件包括侧重构建、持续集成和测试自动化的源代码管理流程和工具，测试自动化包括测试执行、测试结论生成和测试报告生成。如上所述，该工具链还无缝集成了需求管理方面的所有工具。

构建一套集成、高度自动化的工具链可在开发过程中带来极大益处。例如，可降低复杂度从而提升内部效率，可使用来自外部构建

总体而言，引入标准化、最先进的开发工具链，是释放自动化测试和敏捷方法30%至40%效率潜力的关键推动力。这种性能水平足以使表现最佳的企业脱颖而出。车企应将这种软件开发工具链视为高效率组织的核心部分，这类组织都支持连续集成和标准API的使用。此类工具链还可确保整个开发过程中软件与硬件开发工具之间高效的自动化接口。

同时，工具链还为若干流程步骤的自动化（例如测试运行）提供了机会。总体而言，目标就是加快开发速度，促进尽早测试。

通常，用到的工具链和工具的数量会增加到难以管理的水平，从而降低透明度。为避免此问题的出现，经验丰富的工具链管理人员应定期检查工具状况，必要时采取纠正措施。

使转型成为现实

许多组织已开始认真采取行动，希望从软件上获取价值，从而抓住电动化、网联化、智能化、共享化这个“新四化”趋势带来的机遇并从中受益。典型的转型之旅始于某个单一挑战或主题，例如恢复软件项目的即时补救措施、软件“自研或外包”决策或是软件开发绩效方面的重大预期变化。但是，大多数企业发现，单一面向的干预带来的影响非常有限，尤其是当企业缺乏持续改进所需的关键基础时。因此，要建立世界一流的软件开发能力，企业应采取端到端的转型视角。考虑到各自现状的不同，企业可能以不同的方式进行转型。斟酌初始转型侧重点时，可以选择以下的某个切入点：

“是什么”：产品架构和“自研或外包”决策

目的是建立一套目标软件架构，既有具体的内控点，又有外部合作生态圈，从而可实现跨平台的高效且有竞争力的软件交付（请参阅A和B3部分）。

“如何做”：开发效率提升和软件开发方法重点是利用软件开发的关键效率杠杆（包括敏捷研发、持续集成）和自动化测试，提高软件研发的效率（请参阅C和D2部分）。

“在哪里做”：以合理的成本获得人才

面临产能不足、无法有效提升软件产出的企业可先从该主题切入。解决方案包括改善其业务足迹、使组织对软件人才更具吸引力，从而优化人才和研发成本（请参阅B2部分）。

“如何设置”：组织与治理

另一个起点是定义最佳的组织架构，明确“框线”图架构，并说明具体的运营模式，包括绩效引导和绩效管理，从而加强软件交付（请参阅B1和D1部分）。

要克服汽车行业当前的软件复杂性和开发效率难题，就需要对汽车软件研发进行全面的转型。CTO和CEO必须将这一挑战作为其日程上的重中之重，并立即加以应对，才能在当前的行业环境中保持竞争力和成功的市场地位，而且他们还应为更大规模的转型做好准备，因为解决与软件组织及其底层运营模式有关的所有问题可能需要耗时数年之久。

Ondrej Burkacky 是麦肯锡全球董事合伙人，Dominik Hepp 是麦肯锡全球董事合伙人，均常驻慕尼黑分公司；Johannes Deichmann 是麦肯锡全球董事合伙人，常驻斯图加特分公司；Stefan Frank 是麦肯锡咨询顾问，常驻汉堡分公司；André Rocha 是麦肯锡全球董事合伙人，常驻东京分公司。

作者感谢Silviu Apostu, Georg Doll, Julia Gößwein, Anna Gomulec, Virginia Herbst, Shannon Johnston, Maximilian Lemmens, Henrik Rochlitz对本文的贡献。

设计：麦肯锡全球出版部
麦肯锡企业版权所有©2021年。保留所有权利