

面向服务的理念在汽车电子软件架构中的实现路线研究

朱元¹ 苏子钧¹ 徐世寒¹ 王恩东² 刘振东²

(1. 同济大学汽车学院, 上海 201804; 2. 一汽解放汽车有限公司商用车开发院, 长春 130000)

【欢迎引用】朱元, 苏子钧, 徐世寒, 等. 面向服务的理念在汽车电子软件架构中的实现路线研究[J]. 汽车文摘, 2024(3): 21-30.

【Cite this paper】ZHU Y, SU Z J, XU S H, et al. Research on the Implementation Route of Service Oriented Concept in Automotive Electronic Software Architecture[J]. Automotive Digest (Chinese), 2024(3): 21-30.

【摘要】为了适应软件定义汽车的发展趋势并解决面向服务的架构(SOA)在汽车行业应用的适应性调整问题, 首先总结了在汽车架构开发中应用SOA的环节及其优点; 其次, 以自动紧急制动系统软件架构开发为例, 提出一种VSOA实施路线, 提出一种实施路线需求分析流程, 并分析了服务划分原则, 指导了服务逻辑架构和软件架构的建立; 最后, 根据服务逻辑、软件架构、架构前瞻性, 为分析案例建立一种域集中硬件通信架构, 旨在促进汽车电子软件开发中对SOA的理解和应用, 为实现更灵活、可扩展的汽车软件系统提供有益的经验 and 指导。

关键词: 面向服务的架构(SOA); 实施路线; 汽车电子软件架构

中图分类号: TP311.131; U46 文献标志码: A DOI: 10.19822/j.cnki.1671-6329.20230269

Research on the Implementation Route of Service Oriented Concept in Automotive Electronic Software Architecture

Zhu Yuan¹, Su Zijun¹, Xu Shihan¹, Wang Endong², Liu Zhendong²

(1. School of Automotive Engineering, Tongji University, Shanghai 201804; 2. Commercial Vehicle Development Institute, FAW Jiefang Automobile Co., Ltd., Changchun 130000)

【Abstract】In order to adapt to the development trend of software-defined vehicles and solve the problem of service-oriented architecture, the application steps and advantages of SOA in the automotive industry is first summarized. Secondly, taking the development of software architecture of automatic emergency braking system as an example, a VSOA implementation route and an implementation route requirement analysis process are proposed. Furthermore, the service division principle is analyzed, which guides the establishment of service logic architecture and software architecture. Finally, according to the service logic, software architecture and the perspectiveness of architecture, a domain-centralized hardware communication architecture is established, which aims to promote the understanding and application of SOA in automotive electronic software development, so as so provide useful experience and guidance for the realization of more flexible and scalable automotive software systems.

Key words: Service - Oriented Architecture(SOA), Implementation route, Automotive Electronic Software Architecture

0 引言

在汽车行业电动化、智能化、网联化背景下, 整车软件规模和复杂度进一步提升^[1-5], 对通信网络数据量、数据速率和实时可用性提出了更高的要求^[6]。汽车将成为高效的数据交互中心, 并最终演变为更大的移动网络组成部分^[7-8]。以车辆为载体实现这些功能, 意味着电子电气架构必须进行范式转变。航空业率先验证了从分布式架构到集中式架构的可行性和优

点^[9], 汽车行业也已经制定从分布式架构到集中式架构的发展路线, 并有部分整车企业和供应商已经将集中式解决方案商业化^[10]。中央计算+区域控制是当前汽车行业公认的集中式架构发展方向, 能够最大化解耦数据发送方和接收方的依赖关系, 这也为引入面向服务的架构(Service Oriented Architecture, SOA)建立了良好的实施环境^[11]。近年来, 汽车面向服务的架构(Vehicle Service Oriented Architecture, VSOA)能够集中对软件架构和通信方式进行优化。尽管SOA理念

在汽车行业的重要性已经逐渐被相关研究人员和从业者接受,但对于VSOA思想未形成统一的理解,在具体实施上并未形成标准、具体的解决方案。IT行业的服务理念并不能直接适用于多平台异构的汽车系统,对于服务粒度的定义也需要根据系统需求、车辆功能域和具体功能进行具体分析。服务内容和组成元素需要与电子软件架构建立起确定性的映射,这就要求服务和整车架构在设计过程中应该相互反馈。

本文对汽车领域SOA进行了解读,并分析VSOA的特点及其对行业产生的影响。提出了VSOA的标准化开发路线。以建立一个自动紧急制动系统为例,分解开发路线,深入分析汽车电子软件领域中面向SOA的实施方法。分析汽车电子软件系统的特点,展示了服务粒度划分原则和软件架构建立路线,并以软件架构为例设计了决策时延评价服务。基于对数据解耦、服务实现、通信网络复杂度等多方面考量,设计了一种域集中型环网通信架构。

1 相关工作

在软件定义汽车的发展趋势下,SOA成为了电子软件架构的主要模式^[12]。SOA具有清晰的软件层次结构和可伸缩性^[13-15],使软件质量、电子软件架构安全性、软件开发速率、软件远程更新、车内/外信息解耦等多方面得到了优化和敏捷性的提升。SOA借助于中间件能够有效提高服务的标准化、复用性及互操作性。2011年,BMW集团在面向信号的软件中间件—AUTOSAR Classic Platform(CP)基础上结合以太网通信技术,将信号转化为服务开发出面向服务的通信中间件SOME/IP(Scalable service-Oriented Middleware over IP)协议,并于2013年纳入AUTOSAR 4.1规范^[16]。为适应和促进车辆大数据处理性能与高带宽数据传输,AUTOSAR组织在2017年发布了面向服务的中间件AUTOSAR Adaptive Platform(AP)^[17],借助通信中间件SOME/IP协议完成AP和CP的服务互通,进一步提高整车数据解耦。并在2018年将数据分发服务(Data Distribution Service, DDS)纳入AP规范^[18],丰富面向服务通信的实现方案。同年,Edag Group指出将以太网作为域集中架构的主干通信网络有利于车辆全生命周期实现动态软件更新和第三方服务的部署^[19]。

在面向服务的软件架构方面, Lotz^[20]等采用SWOT分析方法,辩证地探讨微服务在汽车系统中的应用,并对面向服务架构在开发、测试等环节的优缺点进行

讨论。Vetter^[21]等基于车机游戏系统的实现,介绍了面向服务的软件开发流程,讨论了企业中软件开发的标准流程和后续服务更新,但未形成系统性的理论和指导方案。Valls^[22]等基于垂直设计方法,开发了中间件iLAND,能够支持面向服务的分布式实时系统有限时间重构。Harald^[23]和Kevin^[24]等采用PREEvision和Ptolomy II实现了电子电气架构模型从上层软件建模到底层硬件实现细节的动态仿真。付朝辉和王华阳^[25]从用户需求出发,指出功能架构建模能够将功能颗粒度细化,并快速封装为原子化服务,而功能依赖关系可以作为服务订阅和发布机制的基础。Marc^[26]等分析了面向服务和面向信号的架构的优缺点,并提出面向服务的架构具有可移植性较强的优点,为实现可重构模块化车辆提供了极大的便利性。Alexandru^[27]等基于需求分析建立面向服务的自动驾驶软件架构生态系统,并通过英飞凌Aurix控制器和电脑之间的消息交换验证了互操作性,并测试了端到端延时。Peter^[28]等研究了22个用于电子电气架构不同阶段开发的工具,并从功能性、兼容性、可用性、分发性4个方面详细分析了各工具的特点,为电子电气架构开发提供了有力指导。

SOA相比较传统面向信号的通信模式,主要区别在于通信路径可以实现动态配置,避免节点加载不必要的数据流,能够支持AP和CP之间的互操作性^[4]。并且对实现方式和通信载体都没有硬性要求,SOME/IP或DDS^[29]是当前主流的实现方式,而传统的CAN总线难以承载上述协议,因此现阶段将以太网作为通信载体。SOME/IP和DDS不仅是通信协议,同时也是通信中间件。SOME/IP被认为是一种非常有前途的SOA通信中间件,但它不包括任何保护应用程序和传输数据免受恶意攻击的安全功能^[30]。而DDS大量的服务质量(Quality of Service, QoS)机制为数据访问、加密、认证等多方面提供安全保障^[31]。另一方面,大量的QoS也提升了DDS体量,必须裁剪后才能用于AP平台,在CP平台需要借助复杂驱动进行定制化开发,从应用角度来说违背了AUTOSAR和面向服务中可重用性的初衷。2者都支持远程过程调用(Remote Procedure Call, RPC),提高服务的重用性和灵活性。

2 汽车电子软件领域SOA概述

2.1 服务的概念

面向服务的体系结构由服务、执行服务的机器以及连接这些机器的网络组成^[32]。服务的概念来源于互

联网,其本义是将应用程序功能抽象成能够作为与服务消费者相关粒度发布的服务集的形式,来提供和使用策略、实践和框架。使用单一的、基于标准化的接口形式从实现中抽象出可以调用、发布和发现的服务^[31]。

一个完整的服务应当具备服务本身(服务的内容)、服务接口以及构成服务的相关角色。服务的内容是一个离散的功能单元,根据服务定义粒度的不同能够实现服务的独立执行和远程访问;信息通过服务接口与外界建立联系,但服务接口与底层通信介质无关;在图1中,SOME/IP实现面向服务的通信展现了一个服务交互的角色集合,包括服务提供方、消费方和代理方。服务的提供方通过代理方注册服务信息,而服务的消费方在代理方进行服务查找和订阅。一旦服务的消费方和提供方成功匹配,就能够进行有效的通信。这种服务交互的方式通过引入代理方,实现了服务的动态发现和订阅机制。服务提供方将服务信息注册到代理方,服务消费方通过代理方查询并订阅所需服务。这个过程的协同性和灵活性使得系统更加适应复杂多变的汽车电子系统中的服务交互需求。这一体系结构使得服务的提供方和消费方能够在运行时动态地协商和建立连接,为整个系统的可扩展性和可维护性提供了坚实基础。代理方的引入不仅简化了服务交互的管理,还提高了系统的可靠性和灵活性。这种角色集合的协同作用使汽车电子系统的服务通信更高效、更智能。

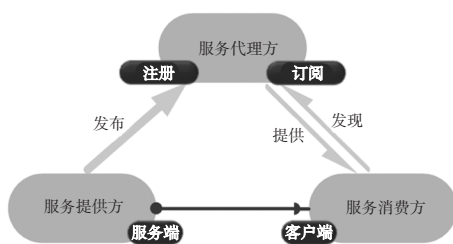


图1 SOME/IP通讯服务中包含的角色

2.2 汽车软件SOA的特点

SOA是一种面向服务的架构,它通过标准化的服务接口、松耦合的服务机制以及可扩展性的服务特性,使得软件具有高度灵活性。在汽车领域,SOA的核心是通过标准化、独立性、松耦合使软件具有高度灵活性。标准化是指每个服务之间应具有清晰的边界,并保留标准化接口以便其他模块在进行功能变更或升级时进行订阅;独立性是指每个服务之间应相互独立且唯一,避免软件在实现任务层面的冗余;松耦合是指服务应独立于车型、硬件平台,甚至于操作系统和开发语言,提高软件开发的效率及灵活性^[34]。

SOA的优势在于提高了业务灵活性、上市速度、可复用性、可伸缩性、安全永续性以及改善了业务与IT之间的协作。SOA的核心要素是服务,每个服务组件具备独立的功能,且可被复用;服务组件之间的接口遵循统一标准,可互相访问,可组合扩展。

AUTOSAR组织在面向信号的CP中引入服务的思想,实现了接口标准化。SOME/IP协议在CP的基础上引入面向服务的通信理念,实现信号到服务实例的转换。在面向服务构建了AP之后,CP和AP能够以统一的面向服务的通信中间件实现数据交互。

在服务的独立性和松耦合方面,AUTOSAR平台的软件分层结构降低了软件与软件间、软件与硬件间的耦合度。在CP中,软件组件(Software Component, SWC)之间的数据交互通过虚拟功能总线(Virtual Functional Bus, VFB)实现,双方只需要留出标准接口,不需要知道其他信息。在AP中,自适应应用运行时环境(AUTOSAR Runtime for Adaptive application, ARA)为SWC预留了服务集合接口,上层应用通过调用ARA服务实现业务内容。算法主体部署在完成接口配置的SWC中,算法工程师能够将工作重点集中在算法内容的开发。在通信过程中,SOA通过通信中间件在系统启动时建立通信路径,调用底层服务定义数据序列化。与面向信号的通信方式具有静态的通信矩阵不同,服务可以动态的建立,不会对其他服务交互产生干扰。在冗余设计场景下,一个客户端具有多个相同的服务源,当原始服务源失效时能够通过服务代理获取替代服务端,提高失效安全性。

面向SOA的软件架构开发,利用服务的高内聚和低耦合,实现软件的全生命周期管理。服务的开发、部署、测试、标定和升级,由独立的团队和服务接口配置的SWC完成^[35]。团队只需在开发阶段,与相关的服务团队沟通,定义服务内容和依赖关系。后续的工作都在团队内部进行。如图2所示,功能模块分解为粗粒度和细粒度的服务,适应不同产品的需求变化。新增细粒度服务,不影响原有服务,便于软件组件的开发。功能升级和扩展也更高效,重用原有资源^[36]。

图2中原子服务的范围由对应的硬件功能决定,在完备的原子服务集合上,合理的组合与划分服务边界,能够实现服务高效复用和灵活的功能分配。组合服务集合中的角色不限于原子服务,也不限于服务部署位置。原子服务可以在独立的测试环境中完成开发-测试-反馈闭环,减少了对其他组件的依赖。原子服务综合到组合服务、应用服务中进行测试,有利于

故障发现与定位,提高测试效率和全面性。

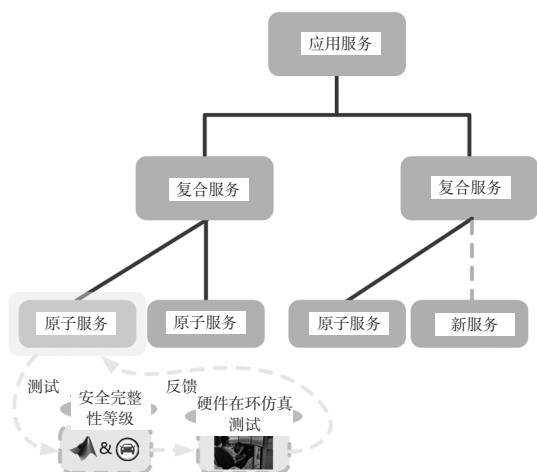


图2 服务粒度划分

根据业务功能的定义可以衍生出流程服务的概念,流程服务是组合服务的一种,它基于服务逻辑关系或业务内容,系统调用多个原子服务及组合服务。

3 SOA在汽车中的实现

面向服务的架构可以通过自上而下/自下而上2种建模路线实现。自上而下的路线需要结合功能特性以及系统的需求、用例和逻辑功能架构等,以系统功能的描述手段作为输入,指导软件架构和硬件架构的建立;自下而上的路线以原有架构为基础,分析服务实现方案,并在服务开发到部署的过程中对原有架构进行适应性改动。

3.1 实施路线图

自上而下/自下而上的建模路线并非是对立的,在建模过程中可以灵活运用2种建模路线,实现“中间相遇”,平衡需求实现和架构资源。本文基于 SOME/IP 实现面向服务通信的方法论,根据面向服务的架构开发经验,提出如图3所示的SOA开发路线。该路线实现了需求到电子软件架构的转换,具有用例驱动和面向服务2个特点,前者指导需求分析,后者指导软件架构建模。用例驱动是指从系统外部观察系统的使用场景、使用方法、功能体现等方面,建立起需求和系统功能逻辑之间的追溯关系。

本文将采用PREvision开发一款AEB系统的电子软件架构。PREvision是一款专业的汽车电气/电子(E/E)系统工程工具,由Vector Informatik公司开发,主要用于支持整个汽车E/E架构的设计、开发和验证。PREvision不仅能够建模整个E/E系统,还能够针对通信协议进行配置管理,包括对从CAN、LIN到现

代的以太网通信,以太网通信中目前支持SOME/IP的灵活配置,使工程师能有效管理系统中不同部分之间的通信^[37]。因此,本文选择PREvision来进行SOA建模。将上述SOA方案解构为需求分析、服务逻辑分析、软件架构建模、硬件及通信架构建模4个方面分析面向SOA的实施路线。

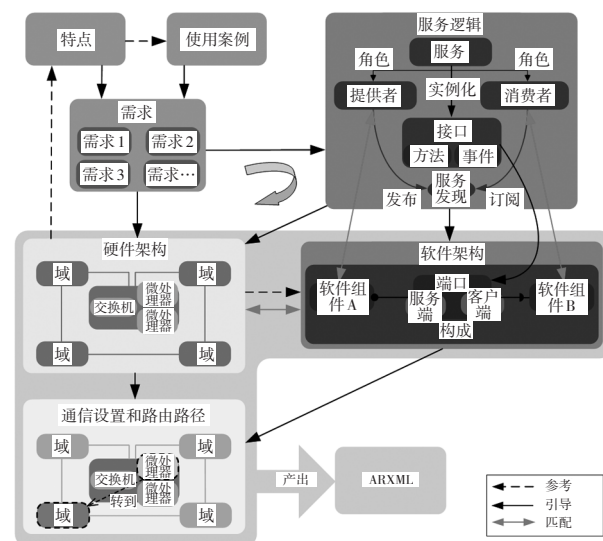


图3 SOA开发路线

选择AEB系统进行建模是因为该系统包含人机交互、环境感知、底盘控制等方面,能够涵盖车辆高级驾驶辅助系统、车身、底盘3个主要功能域。软件平台需要AP和CP的支持,包含了面向服务的软件架构和面向信号的软件架构。本文以该系统为例,在建模前对AEB系统提出以下假设,简化建模环节:

- (1)假设硬件架构末端节点(部件)均为电子控制单元(Electronic Control Unit, ECU),并且ECU中能够部署CP平台;
- (2)假设障碍物为车辆时,T-Box能够获取C2C信息;
- (3)假设AEB系统默认开启。

3.2 实施方案—AEB系统SOA实施方案

需求的提出到需求的实现需要经过3个阶段:需求生成、需求分析、确定需求。需求主要来源于用户调研、用例分析、竞品分析,以及从功能安全和信息安全的角度对系统功能提出具体要求。采用专业语言对需求进行描述,研究当前需求的前置需求和子需求,挖掘潜在需求。在需求分析阶段,从安全性、用户偏好、竞品热度等方面对需求进行优先级排序,保证高优先级的需求首先得到实施。根据外部对需求的反馈评估需求的可实现性,将需求定位到功能实施实体或对应的实施环节中,分析实施实体和实施环节是

否存在潜在需求。最后确定需求内容,指导服务边界、电子软件架构的建立。

根据所提出需求分析路线对AEB系统展开需求分析,确定出以下需求:

(1)应使用多传感器融合来感知前方的障碍物。

a. 视觉信号可以通过传感系统获取。

b. 传感系统应配备中程毫米波雷达。

c. 可以识别障碍物的类型。

d. 如果障碍物的类型是车辆,则传感系统可以从T-Box获取车对车(C2C)信息。

e. 无论可以获得什么C2C信息,都可以通过传感融合来计算相对速度和距离。

(2)系统根据环境信息和主车辆信息进行不同级别的响应。

a. 车门未关闭或乘客未系安全带时,不得启动制动。

b. 自动紧急制动(AEB)和制动辅助需要ESC和ABS的支持。如果ESC关闭或ABS出现故障,则不应启动制动器。

(3)系统应具有安全距离报警模式(需求2的扩展)。

a. 只有当速度大于65 km/s时,此模式才能激活。

b. 当车辆间距小于安全距离时,激活报警。报警应包括语音提示和仪表盘提示。

(4)系统应具有制动辅助模式(需求2的扩展)。

a. 当前制动力不足且速度大于30 km/s时,制动辅助启动。

b. 制动辅助模式应减少对驾驶员的干扰,提高非致敏性。报警仅包括仪表盘提示。

(5)系统应具有AEB模式(需求2的扩展)。

a. 当系统检测到碰撞风险且速度大于30 km/s时,激活此模式。

b. 此模式下的报警应为高强度报警。报警应包括语音提示和仪表盘提示,并具有方向盘抖动。

c. 此模式激活时,制动灯应点亮。

完成需求分析后,需要根据需求内容分析服务逻辑,并建立服务逻辑结构。从需求中能够得知AEB系统需要感知设备获取环境信息,并结合ESC、ABS、制动踏板等提供的主车状态信息进行系统介入决策。介入时需要座舱模块进行介入提示完成人机交互,同时需要获取车门、安全带状态等车身信息,判断是否满足制动实施条件。制动阶段需要ESC&ABS实现AEB计算出的期望制动力。首

先根据以上信息构建出系统工作的逻辑草图如图4所示。

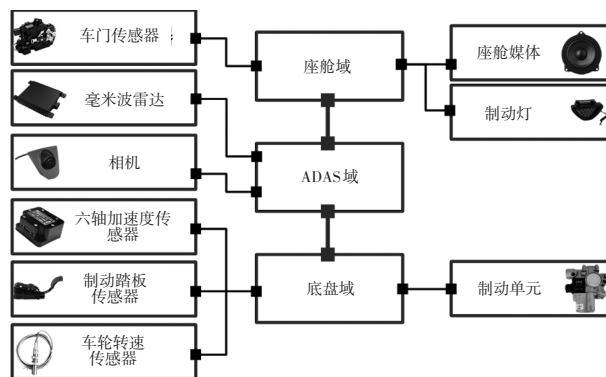


图4 AEB系统逻辑草图

底盘控制算法具有硬实时要求,部署在面向信号的AUTOSAR CP平台上。ADAS域和座舱域需要大数据并行处理,部署在AUTOSAR AP平台上。CP平台软件层可以沿用面向信号的设计方法,也可以面向服务重新构建,设计服务架构时需要考虑服务和信号的转换。

在信号服务化的过程中应当注意服务的内容是由信号抽象而来,还是由对象所提供的响应抽象而来。比如,将制动信号抽象为服务,服务的提供方是制动信号的发送方,消费方是制动信号的接收方(制动器);而把制动这一动作抽象为服务,服务的提供方是制动器,制动信号的来源请求制动器提供服务。2种方案从原理上而言,需要不同的服务实现方式(方法或者事件)。从具体实现上而言,底盘相关模块采用信号抽象为服务的方式更符合控制系统行为,也能有利于软件开发人员逻辑展开。

服务粒度的定义可以来源于需求,以ADAS域中的服务分析,需求3、需求4、需求5回答了需求2对于AEB系统应具有不同工作模式的描述。将ADAS服务拆分为传感器融合、车辆状态计算与决策、AEB控制系统3个子服务,需求3、需求4、需求5对应着AEB控制服务下的细粒度组合服务。需求3中对安全距离计算提出了需求,安全距离计算包含在车辆状态计算与决策服务中,可以通过给定的输入返回确定的计算结果,因此可以将安全距离计算定义为原子服务。即一个应用服务除外部输入服务以外,如式(1):

$$\begin{cases} S_p^n = \{S_m^1, S_m^2, \dots, S_m^{n_1}, S_c^1, S_c^2, \dots, S_c^{n_2}\} \\ S_c^v = \{S_m^1, S_m^2, \dots, S_m^{n_2}, S_c^1, S_c^2, \dots, S_c^{n_2}\} \end{cases} \quad (1)$$

式中: S_p^n 为应用服务, $S_m^{n_1}$ 为 S_p^n 下的元服务, S_c^v 为 S_p^n 下的组合服务, $S_m^{n_2}$ 为 S_c^v 下的元服务, $S_c^{n_2}$ 为 S_c^v 下的子

组合服务。

根据服务的粒度,子组合服务下可以存在更细粒度的组合服务和元服务。服务的粒度并非越细越好,需要根据服务内容进行合理分析,盲目划分服务粒度会给服务架构的全生命周期管理带来麻烦。

根据以上分析可以按照如图2所示粒度划分原则,定义ADAS域中的服务粒度如图5所示。

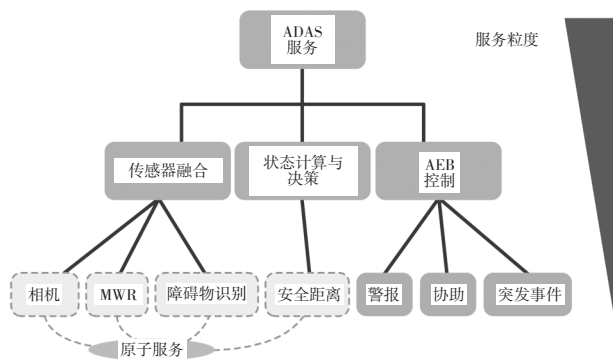


图5 ADAS域服务粒度划分

将ADAS域中服务的分析方法扩展到AEB系统的完全实现,可以得出如图6所示的服务架构。图中CockpitSensorStatus(驾驶舱传感器状态)框中,Cockpit_Ctrl(驾驶舱控制)为服务消费方,CockpitSensorProcess(驾驶舱传感器处理)为服务提供方,虚线用来映射端口角色。

为了更直观的表达,服务架构中重用了部分接口,而在软件架构建模时需要对一些接口进行拆分使服务边界清晰化。

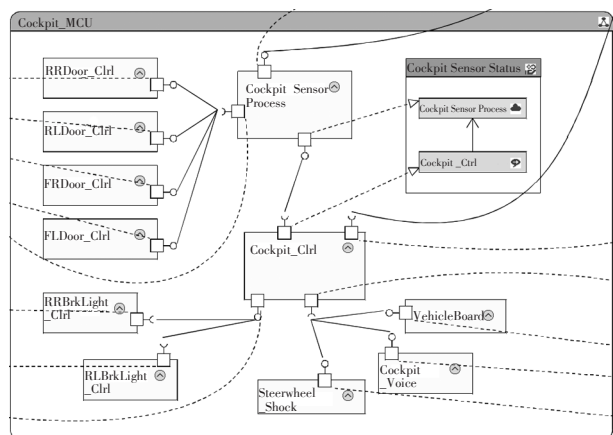


图6 AEB系统服务逻辑架构座舱部分

图6中将服务按照功能域划分并建立服务间的逻辑关系(展示座舱部分),可以看作是逻辑草图(图4)的扩展。服务逻辑架构采用服务语言为数据交互的端口划分了不同的角色,并采用了不同的服务实现方式。例如,系统需要获取语音报警服务时,由Cockpit_Ctrl(驾驶舱控制)向Cockpit_Voice(驾驶舱音量)请求获取服务,语音服务并不需要Cockpit_Voice提供状态反馈,因此以FF-方法的方式实现。而SensorFuse(传感器融合)请求T-Box服务时,需要从T-Box获取数据,因此要以RR-方法的方式实现。SensorFuse获取底盘传感器数据时,服务的内容是ChassisSigProcess(底盘信号处理)的计算结果,因此ChassisSigProcess将目标信号包装为服务,以事件的形式实现服务。图7更加直观的描述了RR-方法、FF-方法、事件的区别。

图7更加直观的描述了RR-方法、FF-方法、事件的区别。

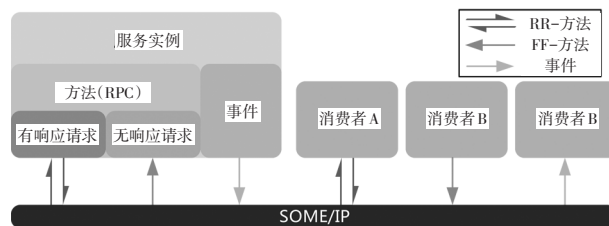


图7 SOME/IP服务实现方法

在SOA软件架构中,弱化了软件组件和硬件单元的映射关系,部署在不同ECU的软件组件可以通过方法实现RPC,提高了软件组件的可伸缩性。

在软件架构中可以将QoS在开发到后续升级的任意环节作为服务被引入软件架构中,这也体现了面向服务的灵活性。端到端延迟一定程度上决定着应用的性能,在AEB系统中出于功能安全的考虑,从传感器传出数据到决策系统提供决策结果这一环节产生的时延对AEB系统的安全性、有效性具有至关重要的作用,因此引入AEB决策时延评价服务。假设车身模块不会干扰决策结果,将上述环节简化为图8,并增加时间戳。图中 T_{s1} 、 T_{s2} 、 T_{s3} 、 T_{s4} 分别为第一个传感器信号发出时、最后一个传感器信号发出时、传感器信号融合结果输出时、决策结果输出时的全局时钟时间戳。

为 T_{s1} 到 T_{s4} 的时间间隔 $t'_{4,1}$ 设置如式(2)所示的约束。

$$\begin{cases} t'_{4,1} = T_{s4} - T_{s1} \\ t'_{4,1} \leq \Delta t_{\max} \end{cases} \quad (2)$$

式中: Δt_{\max} 为设置的时间间隔阈值。

为避免QoS对AEB系统产生影响,式(2)中的约束为软约束。当 $t'_{4,1} > k\Delta t_{\max}$ 时,触发 $t'_{4,1}$ 记录器。记录器采集 $R = [t_{4,1}^{l,0}, t_{4,1}^{l,1}, \dots, t_{4,1}^{l,m}]$,分析 R 的正态分布峰值点坐标 $t_{4,1}^{np}$,若 $t_{4,1}^{np}$ 满足式(3)时,认为AEB服务具有不可靠性,并停用AEB制动服务。

$$t_{4,1}^{np} > k\Delta t_{\max}, (0 < k < 1) \quad (3)$$

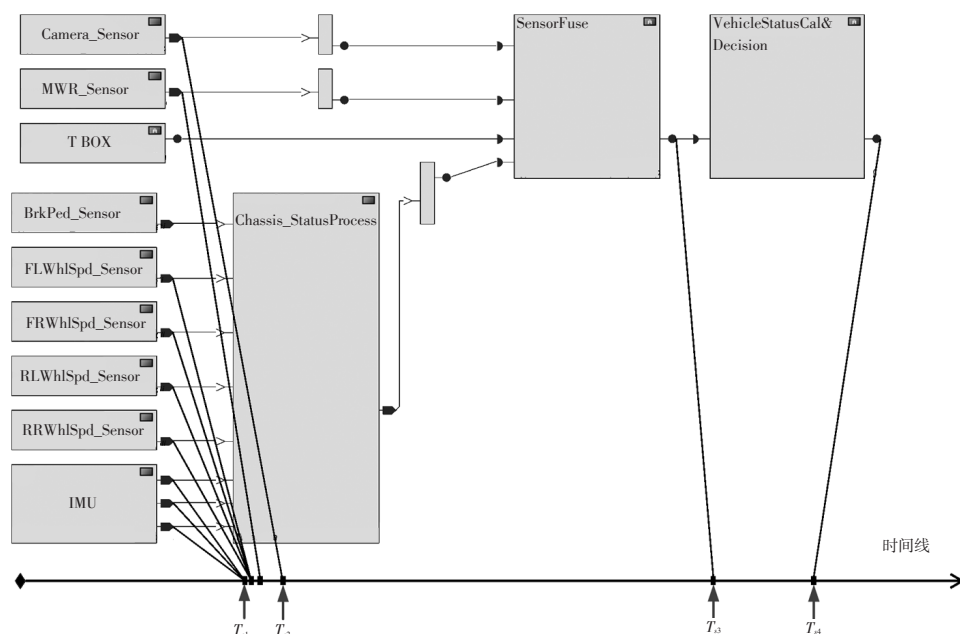


图8 增加时间次序

之后, AEB 决策时延评价服务会一直运行记录器 R , 当连续 δ 个记录周期 R 满足式(4), 则重新启用 AEB 制动服务。

$$\forall t'_{4,1} \in [R_1, R_2, \dots, R_\delta] : t'_{4,1} < \Delta t_{\max} \quad (4)$$

4 域集中式硬件通信架构

汽车电子电气架构在近年来经历了从分布式架构到功能域架构的升级, 许多研究机构正在攻克下一代域集中架构的关键技术。如果说基于功能域的架构相比较分布式架构体现出了高运算能力、高数据解耦程度、简化线束等优点, 域集中架构可以看作在功能域架构基础上再次进行升级, 并确定了以太网作为主干网络的地位。中央高性能计算机作为整车的大脑, 集整车数据中心、中央网关、中央控制单元、中央处理单元于一体, 而区域控制器承担起区域网关、区域配电、区域功能驱动的角色。与仅提供静态通信的 CAN 总线相比, 以太网和 IP 的使用有利于实现动态的面向服务的通信。域集中式架构为 SOA 提供了良好的开发和部署环境, 在 SOA 的加持下能够进一步提升整车数据解耦程度, 实现软硬件的高效复用。

上文在服务架构中将 AEB 服务逻辑划分为底盘域、ADAS 域、座舱域, 并依此将服务映射到软件架构中。底盘域控制算法的特点在于根据确定类型的数据输入, 产生确定性的输出响应。一般具有较大体量的逻辑运算结构, 通常不需要支持复杂的图形界面和图像、信号处理能力, 因此底盘域的算法被广泛布置

在微控制器 (Microcontroller Unit, MCU) 中。ADAS 域需要进行复杂的图像处理、传感器信号融合运算、复杂决策和规划, 往往需要运算单元具有强大的并行浮点运算能力。座舱域需要支持人机交互、图形界面, 同样具有多任务并行处理的特点。因此将 ADAS 域、座舱域布置在微处理器 (Microprocessor Unit, MPU) 中。MPU 和 MCU 及中央交换机可以通过板上总线连接, 实现低延时通信。

分布在车身各位置的传感器、ECU、执行器等由区域控制器采用控制器局域网 (CAN)、串行通信网络 (LIN)、通用串行总线 (USB)、以太网 (Ethernet) 等多种通信方式就近连接 (本文构建的硬件模型中仅采用了 CAN、USB 的连接方式), 区域控制器主要承担区域网关的任务, Ethernet 实现中央高性能计算机到各区域控制器的骨干通信网。根据以上分析, 本文构建出如图 9 所示的域集中环网通信网络。

环网架构的优点在于能够自动选取负载较小的路由路线实现数据传输, 并且在主干网络发生损坏时仍能保证通信系统正常工作。设从中央 Switch 发出的数据流 S_{FL} 、 S_{FR} 、 S_{RL} 、 S_{RR} 的终点分别为对应的区域控制器, 而每段 Ethernet 承载的数据流为 S_1 、 S_2 、 S_3 、 S_4 、 S_5 , 当 Ethernet 发生故障时, 主干通信网上的数据流如式(5)所示。

$$\begin{cases} S_1 = [S_{FL}] \\ S_2 = [S_{FR}, S_{RL}, S_{RR}] \\ S_4 = [S_{RL}, S_{RR}] \\ S_5 = [S_{RR}] \end{cases} \quad (5)$$

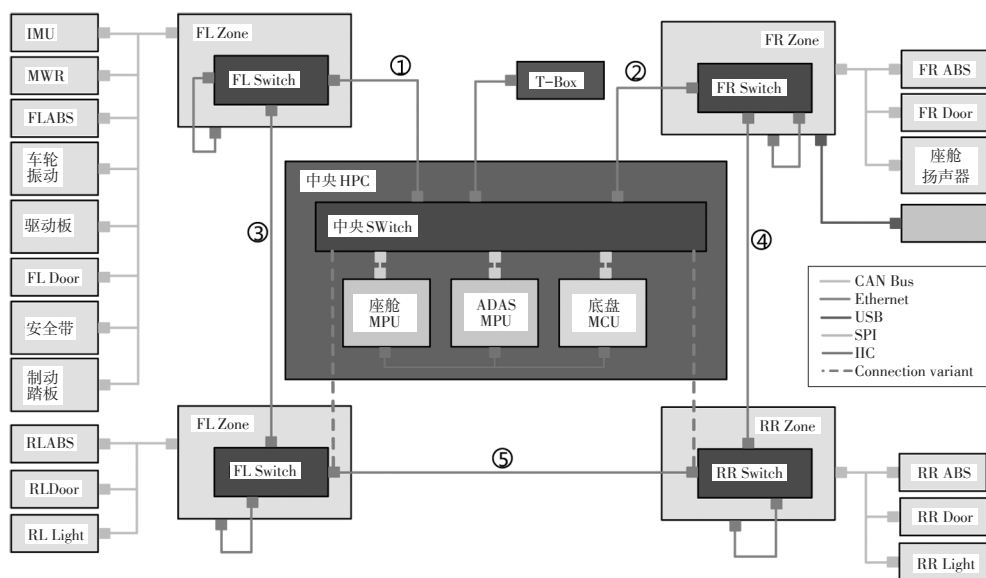


图9 域集中环网通讯网络

图9中还提供了其他的主干通信网连接变体,在采用 Connection variant 后可以形成双环网通信架构,能够进一步降低 Ethernet ①、②的负载,并能够提供更多的信号路由方式。

在进行信号路由之前,需要将SWC映射到域集中硬件通信架构。算法主体将部署到中央高性能计算机,传感器/执行器相关服务映射网络末端节点ECU中。以实现RR Brake Light 服务为例,路由路线起点为座舱MPU,经过中央Switch、FR Switch、RR Switch到达RR Zone,区域控制器承担网关的角色,通过CAN发送点亮右后制动灯的信号。

5 总结与展望

本研究结果揭示了汽车行业在采用面向服务的架构时所面临的挑战和机遇。通过对IT行业和汽车行业服务思想的比较分析,以及对汽车行业SOA特点和优点的梳理,明确了汽车行业在应用SOA时需要进行适应性调整的问题。在此基础上,提出了一种VSOA实施路线,以指导汽车行业如何更好地实现面向服务的架构。通过该实施路线,解决了汽车行业在应用SOA时可能遇到的理论和实际问题,为汽车电子软件架构的发展提供了重要的理论支持和实践指导。针对汽车行业SOA的特点和优点进行了深入梳理,在此过程中,本研究通过需求分析流程、服务划分原则等方面的改进和补充,对前人观点进行了进一步的丰富和发展。因此,本研究结果与前人的一致之处在于对汽车行业SOA的重要性和潜在应用进行了肯定,而不一致之处则在于提出了全新的实施路线和改

进方案,以填补前人研究的空白和不足。虽然本文提出了VSOA的实施路线,但对于其中涉及的关键技术(如全生命周期管理方案、面向服务的通信中间件等)探索不够深入,需要进一步研究和完善。在全生命周期管理方面需要根据汽车电子软件架构的特点设计管理方案,避免架构腐化。在通信方面,面向服务的通信协议如何与时间敏感网络结合以提高实时性,以及需要根据面向服务的通信逻辑定制信息安全方案。通过进一步的研究和探索,可以不断完善VSOA实施路线,提高其在汽车电子软件架构中的实用性和适应性,推动汽车行业SOA的发展和應用。

参 考 文 献

- [1] ZANTEN A T V, ERHARDT R, LANDEFIN K, et al. VDC Systems Development and Perspective[J/OL]. SAE Technical Paper, 1998(2024-02-02). <https://doi.org/10.4271/1980235>.
- [2] BROY M, KRUGER I H, PRETSCHNER A, et al. Engineering Automotive Software[J]. Proceedings of the IEEE, 2007(2): 95.
- [3] VINAY R, HANUMANTHU B, et al. Assessing the Impact of Migration from SOA to Microservices Architecture[J]. SN Computer Science, 2023, 4(5): 577.
- [4] LOB L, MARIANI R, MUBEEN S, et al. Recent Advances and Trends in On-Board Embedded and Networked Automotive Systems[J]. IEEE Transactions on Industrial Informatics, 2019, 15(2): 1038-1051.
- [5] HADI A, MORTEZA H F, ALOIS K, et al. E/E Architecture Synthesis: Challenges and Technologies[J]. Electronics, 2022, 11(518): 518

- [6] NAVALE V M, KYLE W, ATHANASSIOS L, et al. (R)evolution of E/E Architectures[J]. SAE International Journal of Passenger Cars—Electronic and Electrical Systems, 2015, 8(2): 2015-01-0196.
- [7] APOSTU S, BURKACKY O, DEICHMANN J, et al. Automotive software and electrical/electronic architecture: Implications for OEMs[J]. McKinsey Insights, 2019(2023-12-30). <https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/automotive-software-and-electrical-electronic-architecture-implications-for-oems>.
- [8] JIANG S. Vehicle E/E Architecture and Its Adaptation to New Technical Trends[J]. SAE Technical Paper, 2019: 2019-01-0862.
- [9] WATKINS C B, RANDY W. Transitioning from federated avionics architectures to integrated modular avionics [C]// 2007 IEEE/AIAA 26th Digital Avionics Systems Conference. Dallas, TX, USA: IEEE, 2007.
- [10] VICTOR B, GEHAN S, VERA P, et al. Mark Lawford. Making the Case for Centralized Automotive E/E Architectures[J]. IEEE Transactions on Vehicular Technology, 2021, 70(2): 1230-1245.
- [11] TRAUB M, MAIER A, BARBEHON K, et al. Future automotive architecture and the impact of IT trends[J]. IEEE Software, 2017, 34(3): 27-32.
- [12] SCHULTE W R, NATIS Y. "Service Oriented" Architectures[J]. Wiley Interdisciplinary Reviews: Computational Statistics, 1996, 1(1): 101-105.
- [13] NIKNEGAD N, ISMAIL W, GHANI I, et al. Understanding Service-Oriented Architecture (SOA): A systematic literature review and directions for further investigation[J]. Information Systems, 2020, 91: 101491
- [14] PLESSIS J J D, MWALEMBA G, et al. Adoption of emerging technologies into ERP systems landscape: A South African study[C]//IEEE International Conference on Emerging Technologies & Innovative Business Practices for the Transformation of Societies. IEEE, 2016.
- [15] Emadi S, Hanza R H. Critical Factors in the Effective of Service-Oriented Architecture[J]. Advances in Computer Science An International Journal, 2013(3): 14388143.
- [16] AUTOSAR. Release 4.1 Overview and Revision History [EB/OL]. 2015(2023-12-30). https://www.autosar.org/fileadmin/user_upload/standards/classic/4-1/AUTOSAR_TR_ReleaseOverviewAndRevHistory.pdf, 2015.
- [17] AUTOSAR. Adaptive Platform Release Overview[EB/OL]. 2017(2023-12-30). https://www.autosar.org/fileadmin/user_upload/standards/adaptive/1703/AUTOSAR_TR_AdaptivePlatformReleaseOverview.pdf, 2017.
- [18] AUTOSAR. AUTOSAR Adaptive Platform Release Overview[EB/OL]. 2018(2023-12-30). <https://www.autosar.org/standards/adaptive-platform/adaptive-platform-1803/>, 2018.
- [19] MARIO M, GERHARD B, ULRICH B, et al. Service-oriented EE zone architecture key elements for new market segments[J]. ATZechnik worldwide, 2018, 13(1): 36-41.
- [20] LOTZ J, VOGELSANG A, BENDERIUS O, et al. Microservice Architectures for Advanced Driver Assistance Systems: A Case-Study[C]// 2019 IEEE International Conference on Software Architecture Companion (ICSA-C). Hamburg, Germany: IEEE, 2019.
- [21] ANDREAS V, PHILIPP O, HOUSSEM G, et al. Marcel Rumez. Development Processes in Automotive Service-oriented Architectures[C]//2020 9th Mediterranean Conference on Embedded Computing (MECO). Budva, Montenegro: MECO, 2020.
- [22] MARISOL G V, LOPEZ I R, VILLAR L F. iLAND: An Enhanced Middleware for Real-Time Reconfiguration of Service Oriented Distributed Real-Time Systems[J]. IEEE Transactions on Industrial Informatics, 2013, 9(1): 228-236.
- [23] BUCHER H, REICHMANN C, BECKER J. An Integrated Approach Enabling Cross-Domain Simulation of Model-Based E/E Architectures[J]. SAE Technical Paper, 2017: 2017-01-0006.
- [24] KEVIN N, MARCEL R, TREMMEL H, et al. Virtual Verification of E/E Architectures for Secure Automated Driving Functions[C]//2021 IEEE International Symposium on Systems Engineering (ISSE). Vienna, Austria: IEEE, 2021.
- [25] WU H Z, WANG Q, WU Z H, et al. Multi-Material Additively Manufactured Magnetolectric Architectures with a Structure-Dependent Mechanical-to-Electrical Conversion Capability[J]. Small methods, 2022, 6(12): e2201127.
- [26] MARC S, HANNES S, HOUSSEM G, et al. A Comparison of Architecture Paradigms for Dynamic Reconfigurable Automotive Networks[C]//2022 International Conference on Connected Vehicle and Expo (ICCVE). Lakeland, FL, USA: 2022.
- [27] KAMPMANN A, ALRIFAE B, KOHOUT M, et al. A Dynamic Service-Oriented Software Architecture for Highly Automated Vehicles[C]//International Conference on Intelligent Transportation Systems. Auckland, New Zealand: IEEE, 2019.
- [28] WASZECKI P, LUKASIEWYCZ M, MASRUR A, et al.

- How to engineer tool-chains for automotive E/E architectures?[J]. *Acm Sigbed Review*, 2013, 10(4): 6–15.
- [29] Object Management Group. Data Distribution Service[EB/OL]. 2015(2023–12–30). <https://www.omg.org/spec/DDS/1.4/PDF>, 2015.
- [30] ZUO Z, YANG S C, MA B, et al. Design of a CANFD to SOME/IP Gateway Considering Security for In-Vehicle Networks[J]. *Sensors*, 2021, 21(23): 7917.
- [31] RUMEZ M, GRIMM D, KRIESTEN R, et al. An Overview of Automotive Service-Oriented Architectures and Implications for Security Countermeasures[J/OL]. *IEEE Access*, 2020(2023–12–30). https://www.researchgate.net/publication/347178480_An_Overview_of_Automotive_Service-Oriented_Architectures_and_Implications_for_Security_Countermeasures.
- [32] DI N M, SANGIOVANNI-VINCENTELLI A L. Moving From Federated to Integrated Architectures in Automotive: The Role of Standards, Methods and Tools[J]. *Proceedings of the IEEE*, 2010, 98(4): 603–620.
- [33] NIKNEJAD N, ISMAIL W, GHANI I, et al. Understanding Service-Oriented Architecture (SOA): A systematic literature review and directions for further investigation[J]. *Information Systems*, 91(7): 101491.
- [34] ANDREAS P, MARCEL R, DANIEL G, et al. Generic Patterns for Intrusion Detection Systems in Service-Oriented Automotive and Medical Architectures[J]. *Journal of Cybersecurity and Privacy*, 2022, 2(37): 731–749.
- [35] DUC M L, DANIEL L, ARMAN S, et al. An Empirical Study of Architectural Decay in Open-Source Software[C]// 2018 IEEE International Conference on Software Architecture (ICSA). Seattle, WA, USA: IEEE, 2018.
- [36] BEHERE S, TORNGREN M. A functional reference architecture for autonomous driving[J]. *Information and Software Technology*, 2016, 73(5): 136–150.
- [37] SCHAUFFELE J. E/E Architectural Design and Optimization using PREEvision[J]. *SAE Technical Paper*, 2016: 2016-01-0016.

(责任编辑 明慧)

【作者简介】

朱元(1976—),男,同济大学,博士,副教授,研究方向为新能源汽车电机控制技术、汽车电子嵌入式软件、智能驾驶多传感器融合技术。

E-mail: yuan.zhu@tongji.edu.cn

苏子钧(1998—),男,同济大学,硕士研究生,研究方向为汽车电子嵌入式软件。

E-mail: 1070548611@qq.com

徐世寒(1997—),男,同济大学,博士研究生,研究方向为新能源汽车电机控制技术、汽车电子嵌入式软件。

E-mail: sh_xu@tongji.edu.cn

王恩东(1991—),男,一汽解放汽车有限公司,工程师,研究方向为汽车电子嵌入式基础软件技术。

E-mail: wangendong1@rdc.faw.com.cn

刘振东(1992—),男,一汽解放汽车有限公司,助理工程师,研究方向为汽车电子嵌入式基础软件技术。

E-mail: liuzhendong@rdc.faw.com.cn