



面向服务的架构 SOA(Service-Oriented Architecture)是一种软件架构设计的模型和方法。汽车新四化的趋势下,汽车产品正在实现高级电气化功能,例如自动驾驶、V2x、智能座舱等,同时更趋向提升用户体验,如功能的快速更新与升级,提供个性化功能与服务等。

□ 华晨汽车工程研究院 李阳春

面向服务的架构 (SOA) 开发设计方法综述

随着用户和市场多样化服务需求的逐渐强烈,汽车逐渐转变成一个智能终端,除了给用户提供代步功能之外,也提供了其他智能化的体验。而 SOA 这种设计架构的方法正为智能汽车软件提供一种良好的解决方案。不同于传统汽车电子电气分布式架构,SOA 通过标准化的服务接口,低耦合及可组合拓展的特性,结合未来以高性能计算平台即域控制器为核心的集中化电子电气架构,将成为“软件定义汽车”的基础。

SOA 架构设计

1. SOA 定义概述

面向服务的架构 (SOA) 实则是组件模型,其将应用程序的不同功能模块 (即服务) 通过服务间定义好的接口联系起来。接口定义方式独立于实现服务的硬件平台、操作系统甚至是编程语言。这使得构建在各种系统中的服务可以以一种统一且通用的方式进行数据和通信交互,即为服务间的低耦合。当某一个服务变化时,只要接口不变,对其他服务的影响就会很小甚至无感。对低耦合服务的需要来源于应用程序需求,根据业务需求变得更加灵活,以适应不断变化的环境,比如功能变更、迭代和升级等。

在 SOA 架构中,服务是最核心的抽象方式,功能被划分 (组件化) 为一系列低耦合的服务和流程。业务服务相对独立、自包含、可重用,由一个或者多个分布的系统所实现,而流程由服务组装而来。一个“服务”定义了一个与业务相关

的接口,以及约束这个接口的方式。接口和约束接口的方式采用中立、基于标准的方式进行定义,它独立于实现服务的硬件平台、操作系统和编程语言。这使得构建在不同系统中的服务可以以一种统一的和通用的方式进行交互。

SOA 架构中,服务是更高效地利用现有能力满足需求的一种手段,SOA 并不是某一种具体的技术实现,而是一种系统架构的设计思想。SOA 的提出是为了解决随着面临的问题越来越复杂,软件系统变得难以维护、难以扩展及容易出错等问题。

2. 服务与接口定义

(1) 服务定义

目的是定义整车的服务和每个服务的提供者 and 消费者,包括服务 ID 和 VLAN (如果需要)。

1) 元服务: 元服务是最小业务逻辑单元,是对底层逻辑的封装,不具有再拆分的意义和价值;从架构角度,元服务是可通用和可重用的功能单元,构建了整个 SOA 架构的底层基础结构。例如,传感器、执行器等基本接口和基本参数可封装为元服务,将车速、车辆惯性状态等参数封装为车辆状态的元服务,可以对上层服务架构提供支持。

2) 基础服务: 从业务角度,基础服务封装了更多的业务逻辑,即由若干元服务组成;基础服务可以访问和调用元服务以实现更大范围的业务过程。基础服务可足够自治和被重用,在利用元服务的基础上,被重用的汽车业务逻辑可封装为基础服务,例如在利用自车车辆状态服务、雷达

等传感器信息服务以及其他信息的基础上，可构建“环境信息融合”服务，作为基础服务；再如，总线消息的解析和路由（如车身电子服务）、直接与硬件相关的逻辑处理（如音频服务）、上层应用有共同需求的一些基础设施（如电话本或日志服务）。

3) 应用服务：是模型中最上层的服务。从业务角度，应用服务可以包括一个业务过程的全部业务逻辑，封装了与特定业务有关的基础服务；从架构角度，应用服务可以访问和调用基础服务以帮助其解决业务问题。元服务和基础服务更强调架构的灵活性，而应用服务则更强调功能本身，和业务需求有非常紧密的联系。例如，制动能量管理是一个应用服务，它利用车辆状态元服务、环境信息融合基础服务和其他服务，从而实现了通过智能管理降低车辆能耗的功能。也可以需要应用服务之间相互协同实现业务逻辑，例如导航服务。

在设计中，上层服务调用下层服务，下层服务不调用上层服务，这一原则有助于构建清晰简单的 SOA 汽车软件架构（图 1）。参考以上分层模型，可以有效定义 SOA 汽车软件的开发方法，从而有效地发挥和获得 SOA 的优势。

（2）服务接口定义

服务的能力及特征由其服务接口定义（图 2），服务接口的定义决定了服务消费者如何使用服务，服务接口由如下子元素组成：① Properties（Field），描述某个数据，该数据可由服

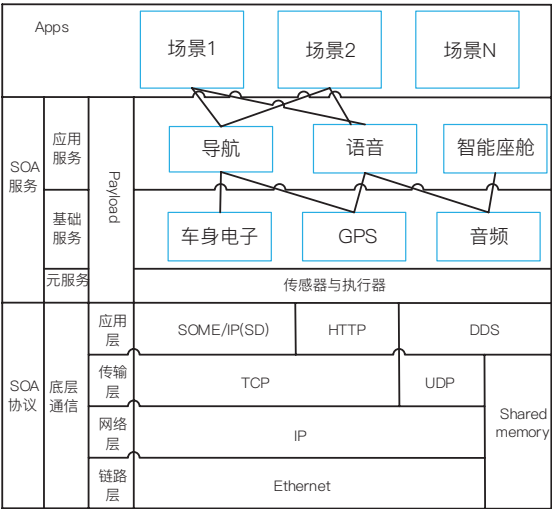


图 1 SOA 服务框图

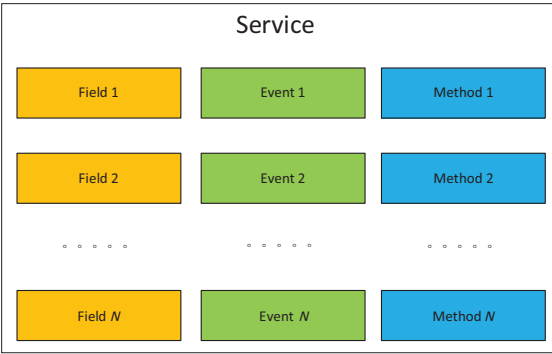


图 2 服务接口定义

务消费者 Client 读取或写入，也可以生成一个 NotifierEvent，如果这个数据发生变化；② Event，当事件发生时，服务提供者 Service 通过 Event 向订阅的 Client 发送数据；③ Method，描述可用的 Operation，可以从其他服务进行调用，方法产生并返回一些数据（R&R），或者在服务中启动一些进程（F&F）。

3.SOA 架构设计原则

SOA 架构设计原则本文概述如下：

1) 从最上层的整车配置分解成对应功能列表，将系统划

分为不同的子系统。

2) 整车功能完全独立不重合，一个子系统只承担特定的一个功能特性，同时一个功能特性也只能在一个子系统中，不能在其他系统中重复定义。

3) 子系统通过接口来交互从而不依赖于其他子系统的内部功能逻辑，即为低耦合。

4) 在功能细节不明确，或者功能不断进化的情况下，需要避免急于对细节进行设计。

5) 服务进行分层管理，将相同类型的服务归纳到相同的服务层，避免将不同类型的服务放到同一逻辑层，尽量做到服务组合，而非迭代继承服务。

6) 在服务层级下，不能跨层调用服务，要保持服务的独立性。

7) 不同服务分层下，避免数据类型格式的转换，比如频繁的物理值和信号值之间转换是必须要避免的。

8) 性能属性代码尽可能独立于应用功能类，否则一旦出现性能需更新时，会影响功能逻辑也需更新，导致架构难以维护与拓展。

9) 在实际开发之前，尽量做好建模和可视化仿真工具分析，提前识别风险和漏洞。

4.SOA 开发设计流程（图 3）

总体而言，通过分层部署的方式，将全局变量的服务统筹提取出来（例如车速、时间、车辆状态等），作为共用的软件模块提供服务，实现特定的基

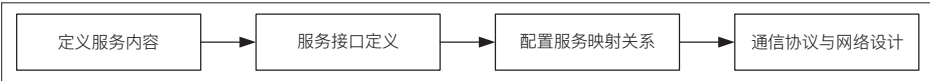


图 3 开发流程步骤



础功能软件接口统一，并可灵活部署。

(1) 定义服务内容

此步骤实际上就是搭建了一个系统功能架构，从整车层面即是按照功能需求定义并划分服务。对于 SOA 中的服务表示了一种独立的功能单元，一个服务可以包含其他子服务单元，使用标准接口进行通信，将内部信息封装成一个黑盒子，实现子服务的重用性。上层服务可以通过该标准接口调用下层服务封装的子服务内容。同时，整体的服务内容可以被操控单元远程访问和独立更改或更新。同时，对于 SOA 来说，需要通过服务编排来定义清楚服务之间的相互关系。

简单地说服务对于智能汽车而言就是定义产品功能，对其中产品的能力 (Product Capability, PC) 进行描述。实现这种产品能力需要从下至上定义硬件抽象服务、平台核心服务、域核心服务、应用程序服务。而每一个服务内容对应着一个或多个实现的软件模块 (Software Capability, SWC)。

产品能力 (PC) 描述了系统所需的一些高级功能。区别于系统设计，产品能力是用来分配职责的，因此很清楚哪个 SWC Module 软件模块 (如摄像头识别模块、雷达识别模块、中央域控制器模块) 应该实现什么。它们在功能设计时由功能负责人识别和请求，如果涉及系统相关功能，则由系统架构工程师直接识别。在确定并决定添加 PC 后，对应的软件模块拥有该 PC，模块所有者负责将其实施到正确的

版本，并在平台的整个生命周期内维护 / 发展 PC。

(2) 定义服务接口

服务接口是一种通信内容定义，其目的在于将服务从功能架构过渡到软件技术架构，且软件模块之间的关系需要被清晰地定义出来，过程中将服务内容封装成相应的接口被实际调用。这种接口定义是独立于通信协议的抽象实体，这种接口可以建立任何两个服务间的通信能力，而使用合适的工具链可以由此生成基于特定协议的接口。

服务接口可分为方法 (Method)、属性 (Property)、事件 (Event) 三种类型。以智能驾驶的一个子功能执行接口服务为例，假设需要获取摄像头传感器探测的环境数据，而需要进行定义的服务接口中方法是要对传感器的参数进行后融合，那么就需要其底层服务提供摄像头处理的基础函数 (如 ISP、深度学习函数、BEV 函数等)。而服务接口的属性则是通过一定的方法操作 (如 get/set) 来获取该服务函数，这种服务属性可以对上层调用的服务部分可见，底层服务有变动上层的调用方式也会随之变动，这种变动所带来的更新会由服务底层决定何时发送给上层调用它的服务单元。

服务接口定义完整后，开发人员可以根据该接口定义对其中的函数进行定义开发了。

(3) 配置服务映射关系

此过程会建立软硬件之间的映射关系，实现从抽象的服务定义到软件层面的推导，从

而方便实现软件驱动或调用硬件实现单元，这种结果是实现服务与中间件或底层硬件 ECU 之间的映射关系。从整个 SOA 的架构模型中我们知道服务需要从通用服务平台开始进行底层驱动，然后对上层传感器执行器的控制管理进行驱动。如是 Adaptive AutoSAR，则 AP 直接支持服务接口，可以直接面向上层应用层，如是 Classic AutoSAR 仍然是对常用的底层应用服务的驱动映射。

(4) 通信协议与网络设计

SOA 架构设计需要强大的环境感知、信息处理、实施决策、控制能力，以及地图、定位、通信、云及大数据等进行系统集成，故车端与云端、车辆与车辆之间、车辆内部的各个 ECU 之间通信的速率和数据量都比传统汽车高出几个数量级，这些需要由多种复杂的硬件、软件和高速通信总线共同实现，并在很大程度上决定智能汽车的功能实现和扩展的可靠性。车载以太网能够很好地解决大数据量的信息交互，整个通信协议的定义包括虚拟以太网 VLAN，以太网交换机 Switch，套接字 Socket，基于 IP 的可扩展面向服务的中间件 SOME/IP，SD 等。而基于 AVB 的下一代协议 TSN (时间敏感网络) 可以提供非常优秀的实时性。当然 TSN 如何应用现在仍然是一个待深入研讨的课题。目前应用的厂商确实屈指可数。

以太网通信设计过程包含对服务实例进行通信协议相关的信息配置。由于 SOA 架构

(下转第 29 页)

表 2 智能网联汽车信息安全相关标准法规

序号	标准号	名称	发布时间 / 状态
1	UN R155	信息安全与信息安全管理系统	2020.6
2	UN R156	软件升级与软件升级管理系统	2020.6
3	ISO SAE 21434	道路车辆 - 网络安全工程	2021.8
4	ISO PAS 5112	道路车辆 - 信息安全工程审核指南	起草
5	ISO 5083	道路车辆 自动驾驶系统的安全性和网络安全 设计、验证和验证方法	起草
6	ISO 24089	道路车辆 - 软件升级工程	起草
7	SAE J3061	信息物理汽车系统网络安全指南	2016.1
8	GB XXXX	汽车整车信息安全技术要求	起草
9	GB XXXX	汽车软件升级通用技术要求	起草
10	GB/T 40855-2021	电动汽车远程服务与管理系统信息安全技术要求及试验方法	2021.10.11
11	GB/T 40856-2021	车载信息交互系统信息安全技术要求及试验方法	2021.10.11
12	GB/T 40857-2021	汽车网关信息安全技术要求及试验方法	2021.10.11
13	GB/T 40861-2021	汽车信息安全通用技术要求	2021.10.11
14	GB/T XXXX-XXXX	电动汽车充电系统信息安全技术要求及试验方法	报批
15	GB/T XXXX-XXXX	汽车诊断接口 (OBD) 接口信息安全技术要求	起草
16	GB/T XXXX-XXXX	整车级信息安全风险评估规范	起草
17	GB/T XXXX-XXXX	汽车信息安全应急响应管理指南	起草
18	GB/T XXXX-XXXX	汽车整车信息安全测试方法	起草
19	GB/T XXXX-XXXX	道路车辆 信息安全工程	起草
20	GB/T XXXX-XXXX	信息安全技术 汽车采集数据的安全要求	审查

要强化顶层设计，通过立法明确各方在信息安全中责任分配，通过发布相关指导性文件、制定政策法规等手段规范行业行为。目

前，我国政策法规及国家、行业标准整体已逐步完善。企业层面需要建立全面的信息安全管理制度和应急响应机制，重视相关硬

件、软件和系统的技术创新，在企业管理、产品开发及系统运营等全过程中，建立网络信息安全保障机制。AI

(上接第 24 页)

中包含多个应用实体之间的多通路通信过程，且这些通信通常是网状通信，因此需要在各个实体节点之间建立中间路由、转化等。区别于传统总线(CAN/LIN)，在软件架构设计过程中，开发人员需要设计具体的服务类型、服务 ID、服务数据类型、服务角色等。

总结

SOA 架构主要优势是可以

很大程度上实现软硬解耦，通过软件升级 OTA 可以更加方便灵活地实现服务实体有效部署在任意的域控制器上，而且可以在售出车辆上调整软件部署或者升级策略。同时，SOA 的衍生功能是可以汽车功能安全方面实现有效的冗余部署。例如，对于安全性要求比较高的智能驾驶功能可以实现双重制动冗余配置，同时在车端布置双重中央域控制服务，确保当一个域控制器失效时，

那么另外一个域控制器上的备用服务实例立刻启动，重新与服务使用者建立连接，以保证功能的正常运转，借此实现冗余机制。当然 SOA 的批产实现现在看来也不是一蹴而就的，还是需要面对很多的挑战，例如复杂度、时间性、网络安全和后面的长期支持，都是主机厂需要考虑的问题。所以 SOA 技术的实现到落地到后期维护现在看来还是需要砥砺前行。AI