

一种能力驱动的开放式航电软件架构

朱剑锋, 洪 沛, 崔 杰, 高忠杰

(中国航空无线电电子研究所, 上海 200041)

[摘 要] 在新一代 IT 技术环境下, 传统单平台“烟囱式”的航电软件架构和开发模式面临着变革的挑战。针对航电软件系统软硬件耦合紧密、规模大、复杂度高的特点, 提出了一种能力驱动的开放式航电软件架构环境, 用于大型复杂航电软件系统的多团队开发。除了软件计算环境 (SCE) 为可移植的应用软件组件提供跨飞机平台的能力外, 该环境还包括一个航电软件架构规范, 保障架构的开放性、可移植性、互操作性等, 以及规范基于模型驱动的软件开发流程, 用于快速部署、升级和替换可移植的软件组件。为了证明该开放式航电软件架构环境的可行性和有效性, 本文结合实际需求进行了实例研究。

[关键词] 能力驱动; 航空电子系统; 软件开发环境

[中图分类号] TP311.5

[文献标识码] A

[文章编号] 1006-141X(2022)02-0033-06

A Capability-Driven Open Avionics Software Architecture

ZHU Jian-feng, HONG Pei, CUI Jie, GAO Zhong-jie

(China National Aeronautic Radio Electronics Research Institute, Shanghai 200241, China)

Abstract: In a new generation of IT technology environment, the traditional single-platform “chimney” avionics software architecture and development model are facing the challenge of changes. Due to the characteristics of tight-coupling of software and hardware, large scale and high complexity of avionics software system, a capability-driven open avionics software architecture is proposed for multi-team development of large-scale complex avionics software systems. In addition to a Software Computing Environment (SCE) which provides capabilities cross aircraft platforms for portable application software components, the architecture also includes a consistent avionics software architecture specification with key quality attributes such as openness, portability, interoperability; and the specification also specifies the model-driven software development process in order to quickly deploy, upgrade and replace the portable software components. To prove the feasibility and effectiveness of the architecture, a case study based on some actual requirements is illustrated.

Key words: capability-driven; avionics system; software development environment

云计算本质上是一种 IT 能力的生产消费范式, 与基于开放标准的移动互联网和物联网应用相结合, 催生了一个大数据时代, 进一步促进以机器学习或

深度学习为代表的人工智能计算能力的推广和广泛应用。在新一代 IT 技术环境下, 传统单平台“烟囱式”的航电软件架构和开发模式也面临变革的挑战。在

收稿日期: 2021-10-24

引用格式: 朱剑锋, 洪沛, 崔杰, 高忠杰. 一种能力驱动的开放式航电软件架构 [J]. 航空电子技术, 2022, 53(2): 33-38.

此背景下, 针对航电软件系统软硬件耦合紧密、规模大、复杂度高的特点, 本文提出了一种能力驱动的开放式航电软件架构, 用于大型复杂航空电子软件系统的多团队开发:

(1) 具有开放、可移植性、互操作性、高可用性和高可伸缩性等关键特性的航空电子软件架构^[1]规范;

(2) 跨飞机平台的软件计算环境;

(3) 基于软件组件和微服务、模型驱动^[2]的软件开发流程, 以便快速部署、升级和替换可移植的软件组件。

本文分析了 ASAAC、ARINC 653、POSIX、FACETM^[3] 等开放标准以及新 IT 技术的相关开放技术规范。跨飞机平台的软件计算环境 (SCE: Software Computing Environment) 基于现有标准和技术, 通过对硬件“虚拟化”、对软件接口标准化以及关注点的分离而构建, 实现了软件架构的开放性和软件系统的松耦合, 并提供基于 SCE 的便携应用软件组件能力, 进而形成了能力驱动的开放式航空电子软件开发方法。组件能力主要包括:

(1) 基于机载实时操作系统的软件操作环境能力;

(2) 具有标准接口的数据输入/输出功能, 可适用于特定的硬件设备, 并与各种消息机制兼容;

(3) 通过标准化接口传输信息的能力, 该接口可与多种传输协议兼容, 实现可移植的软件组件之间的互操作性。

为满足大规模复杂的航空电子设备软件系统的实时性、可靠性、可预见性和可调度性需求^[4], 实现关注点分离、可移植的单元模型之间的数据交换以及数据模型的建立, 本文在规范中针对数据模型和服务组件, 引入并规范了基于模型驱动的软件开发流程。数据模型是一种持续演化改进的数据元仓库, 它需要一种演化控制机制来实现数据模型的管理、数据模型元素的状态监控、管理和演化控制。最后根据实际需求, 利用能力驱动的开放式航电软件架构开发了一个航空电子软件系统原型的试点项目。

1 能力驱动的开放式航电软件架构

能力驱动的开放式航电软件架构由软件计算环境和航空电子软件体系架构规范组成, 如图 1 所示。SCE 为可移植的应用软件组件提供跨飞机平台的能力, 包括平台适配能力、数据传输服务能力、基础服务能力、领域服务能力和应用开发框架; 这些功能通过指定的接口公开给航空电子软件使用。航空电子软件体系架构规范详细说明了构建软件产品时, 对模型以及包含数据模型的应用程序组件集成过程。同时规范中约束软件开发流程, 为软件计算环境提供验证过的数据模型和服务组件。软件开发流程中有三个角色, 管理员管理数据中心和应用中心, 分别存储模型数据和应用组件; 开发人员创建模型和组件; 集成商将它们集成在一起。

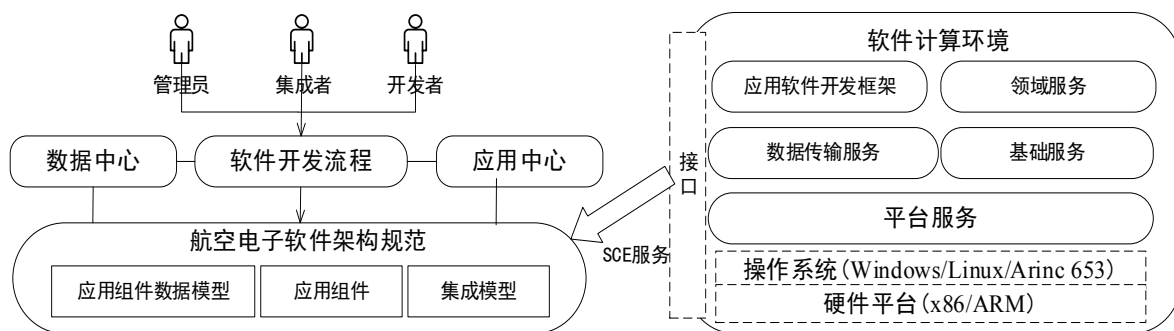


图 1 能力驱动的开放式航电软件架构

2 软件计算环境

能力驱动的开放式航电软件架构软件计算环境建构在 Windows、Linux、ARINC 653 等本地操作系统上, 提供了可移植应用的能力, 包括平台适配能力、

数据传输服务能力、基础服务能力、领域服务能力和应用程序开发框架, 如图 2 所示。通过这些功能, 可以更高效地构建航空电子软件产品, 满足可移植性、兼容性和安全性的要求。

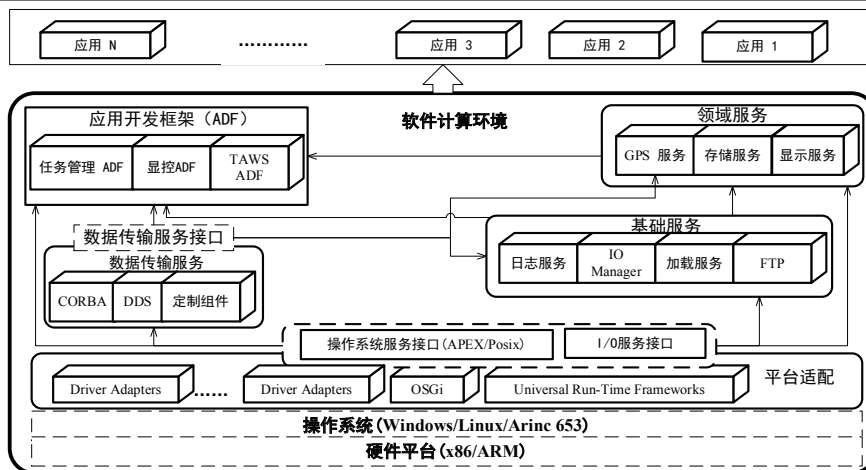


图2 软件计算环境

2.1 平台适配能力

平台适配功能包括驱动适配、OSGi 和通用的运行时框架，为操作系统服务和数据输入/输出服务提供标准化接口。通过平台适配功能，应用程序可以实现与特定操作系统的解耦。

2.2 数据传输服务能力

数据传输服务能力主要通过采用成熟协议的通

信组件来实现，可支持本地通信（共享内存）和远程分布式通信（Socket、DDS）。通过这种方式，建立在数据传输服务能力之上的应用程序仅依赖于通信组件，可完全与硬件平台解耦；而通信组件调用平台适配能力提供的标准化的操作系统服务接口和 I/O 服务接口（如 APEX 接口和 POSIX C 套接字接口等）来处理与底层硬件平台的关系。

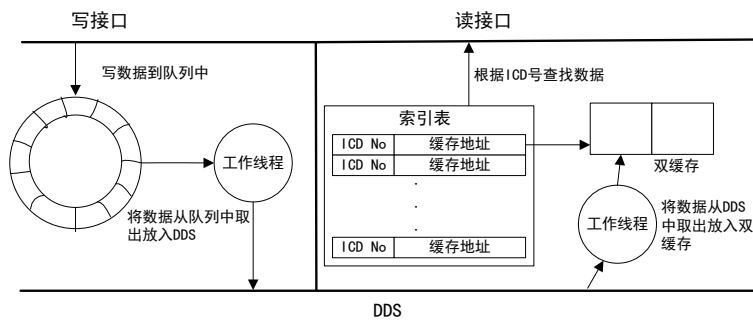


图3 DDS 读写示例

如图3所示，对于写场景，组件首先通过写接口将数据写入队列，然后工作线程从队列中提取数据并发送给DDS；在读场景中，工作线程从DDS中提取数据并将其发送到双缓冲区，其中有一个索引表记录了ICD号及其缓冲区地址，因此组件可以通过读接口根据ICD号查找数据来读取数据。

2.3 基础服务能力

以平台适配能力和数据传输服务能力为支撑，基础服务能力实现与运行时环境相关的数据加载、数据记录、平台维护管理等基础功能，主要包括系统管理、数据加载管理、文件服务、日志服务等；基础服务组件还可以通过直接访问标准的操作系统接口和网络接口来实现特定的平台服务。

基础服务能力主要包括：

- (1) 基于ASAAC标准的系统管理服务；
- (2) 基于FTP/TFTP/SNMP的网络服务；
- (3) ICD到应用数据模型的转化；
- (4) 基于ARINC 615A的数据加载服务；
- (5) 标准化的日志服务。

基于ARINC 615A协议的加载服务提供了每个应用程序目标文件和配置文件的加载服务，并且任何基于ARINC615A协议的设备或系统都可以成功接入系统，完成应用程序或配置文件的升级。基于ASAAC标准的系统管理提供对设备或系统资源的管理能力，软件开发者可以将注意力更多地集中在数据处理、逻辑交互和算法上。根据系统需求，SCE

提供经过验证的基础服务,使得 ARINC 615A 加载服务和 ASAAC 系统管理成为系统中各个子系统或设备的典型配置。

基础服务组件调用标准化的操作系统服务接口和平台适配能力提供的输入输出(I/O: Input/Output)服务接口,以及数据传输能力提供的数据传输服务接口。在相关协议和框架要求的前提下,基础服务能力直接提升系统的专注度,使系统更加专注于任务,使应用程序与特定平台分离。

2.4 领域服务能力

领域服务能力提供特定领域的通用服务能力,如 GPS 服务、显示服务、算法服务等。领域服务组件是一组跨平台、互操作且能够满足航空电子领域任务需求的功能组件;他们调用标准化的操作系统服务接口和平台适配能力提供的 I/O 服务接口,数据传输能力提供的数据传输服务接口,并利用基础服务能力来实现领域功能;它们经过多重验证并整合为可靠的应用程序组件库,以创建潜在的系统功能池。对于各类飞机或直升机的功能设计或能力提升而言,由于领域服务组件库直接面向用户,设计人员可以直接使用相关的能力组件,或基于能力组件进行二次开发,从而大大缩短开发周期。

2.5 应用程序开发框架

应用程序开发框架为应用程序开发时提供主体结构,如显示控制应用的页面调度框架和数据管理的文件系统管理框架等。该框架是从海量领域应用中提炼出来的通用框架,在领域服务能力方面又迈进了一步,它保持了领域应用的结构稳定性,简化了开发过程,如图 4 所示。

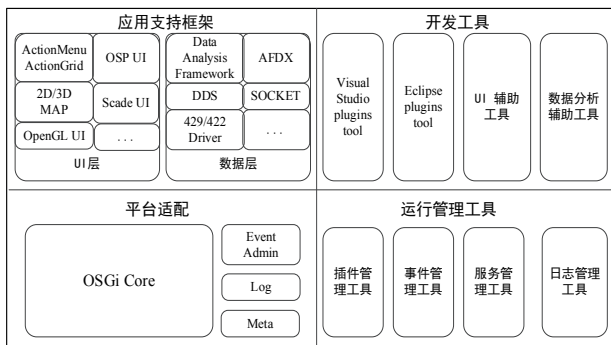


图 4 应用程序开发框架

该框架包括了四部分的内容,每一部分都可以独立支撑应用软件的开发:平台适配能力支撑功能应用,应用支持框架提供显控应用开发的框架,Vi-

sual Studio 插件工具、Eclipse 插件工具等二次开发工具提供便捷的人机界面开发工具,最后运行时管理工具(例如插件管理工具和事件管理工具等)提供二次开发框架,实现运行时管理功能。

3 软件架构参考规范

本文基于 SCE 的软件架构,提出了一种参考规范,该规范指定应用程序应基于 SCE 提供的标准化接口和功能进行开发,以获得可移植性。图 5 显示了软件架构参考规范。

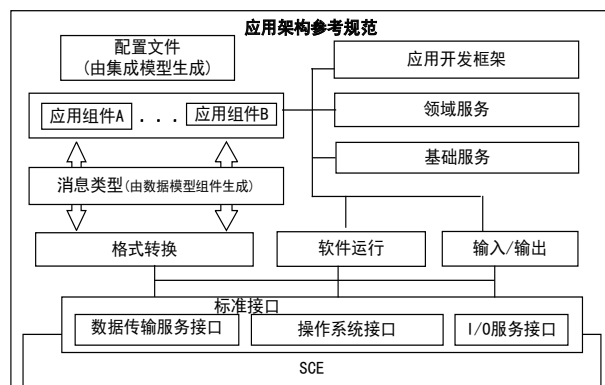


图 5 软件架构参考规范

3.1 仅使用由 SCE 提供的能力

传统航电软件与计算平台的紧密耦合主要涉及 I/O 接口、操作系统服务接口、数据传输服务接口等,应用需要使用且仅使用 SCE 提供的相关接口和服务能力与计算平台解耦。这些能力包括基础服务能力、领域服务能力、应用开发框架等,基于这些能力开发的应用符合能力驱动的开放式航电软件架构的要求,具有良好的可移植性。

3.2 应用组件

经过验证的航电软件组件是航电系统的重要组成部分,为了实现应用组件之间独立于计算平台的互操作性,规范规定组件之间传递消息的接口必须使用 SCE 提供的标准化数据传递服务接口。

3.3 信息的数据类型

传统航电软件应用组件之间的互操作性问题还包括数据传输服务接口使用的数据类型的一致性问题。数据建模可以为应用组件之间的数据传输提供标准化的方法,其从应用层的角度建立语义模型,为传输接口提供规范化的语义信息。数据模型生成的数据消息类型可以保证应用程序组件的互操作性。

3.4 配置文件

嵌入式航电软件大多是根据配置文件来启动的, 比如 ARINC 653 平台的应用程序是根据配置文件来确定分区和时间运行框架的。为了更好地规范软件架构中的应用程序, 软件架构参考规范指定应用程序配置文件由应用程序的集成模型生成, 这也增强了应用程序的可移植性。

3.5 基于模型驱动的软件开发流程

参考规范约束 SCE 环境下的基于模型驱动的软件开发, 限于数据模型和软件组件。确保经过验证后数据模型和组件纳入 SCE 的相关服务中, 后续用户可通过配置文件或者接口调用实现相关功能, 开发流程如图 6 所示。

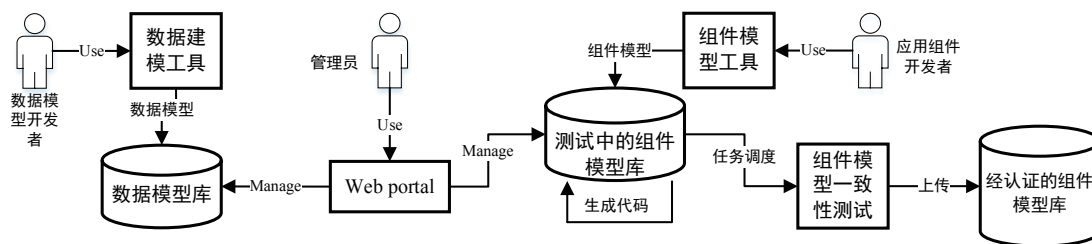


图6 基于模型的开发流程

(1) 数据模型开发者使用数据建模工具开发数据模型;

(2) 数据模型必须通过 OCL 验证, 然后上传到数据模型库;

(3) 应用组件的开发者使用组件建模工具生成组件模型;

(4) 组件模型将上传到待测组件模型库中, 组件模型的组件代码将自动生成。

(5) 任务调度器分配组件代码进行一致性测试;

(6) 如果代码通过一致性测试, 则将其上传到经过认证的组件模型库并为集成商准备。

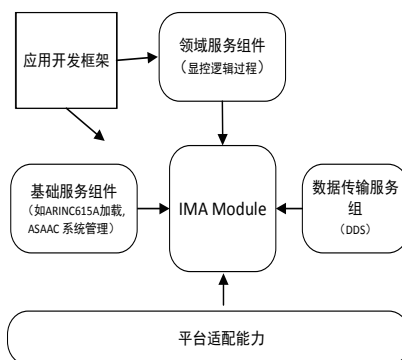


图7 SCE对IMA模块的能力支持

4 案例研究

4.1 案例描述

机载嵌入式显示系统的运行平台由多功能显示器 (MFD: Multi Function Display) 和集成模块化航空电子设备 (IMA: Integrated Modular Avionics) 组成。IMA 模块主要负责显示系统 ARINC 661 应用软件^[5]的逻辑处理, 其接收和处理的数据来自 MIL STD 1553B 总线和 ARINC 664 网络。该项目使用 SCE 为 IMA 开发软件系统模型。图 7 刻画了 SCE 对 IMA 模块的能力支持。图 8 描述了 IMA 软件计算环境。

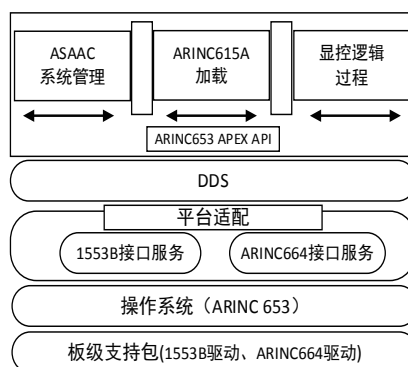


图8 IMA软件计算环境

4.2 能力驱动的开放式航电软件架构的应用

平台适配能力借鉴 FACE 的 IO 标准接口, 使用 Initialize(I/O)、Read(I/O)、Write(I/O) 和 Get_Status(I/O) 等接口。Initialize(I/O) 接口加载 MIL STD 1553 设备通讯表和 ARINC 664 设备网络通讯配置表, 初始化 IO 设备; 应用 Read(I/O) 和 Write(I/O) 接口来实现总

线数据的发送和接收。数据分发服务(DDS: Data Distribution Service)应用原生接口分发各个功能应用的数据,实现数据传输服务功能。

本例中使用的原生操作系统是风河的 VxWorks 653,基础服务组件和领域服务组件以分区应用的形式驻留,其输入输出通信基于标准的 APEX 接口。在外部通信模块的实现中,分区 APEX 接口和 DDS 之间建立了传输接口,该接口基于 VxWorks 653 系

统框架实现。案例中,基础服务组件为 IMA 模块提供系统管理和数据加载能力:系统管理基于 ASAAC 标准实现 BIT 管理、故障管理和日志管理功能;数据加载能力是基于 ARINC 615A 协议的数据加载组件,可以直接添加到基于 ARINC 615A 协议的数据加载系统中,完成软件和配置文件的更新。基础服务组件是一系列完整的服务组件,可以通过配置文件直接形成工作于特定平台的能力支持。

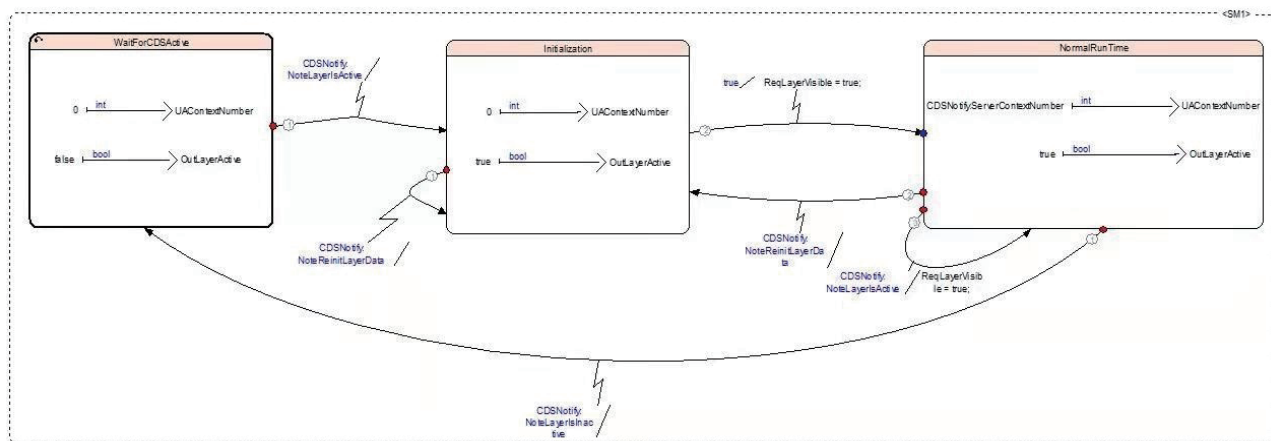


图9 领域服务的逻辑处理模型

案例中,领域服务组件可直接满足 IMA 模块的特定系统需求,如负责显示系统的 ARINC 661 应用软件的逻辑处理能力。图9展示了 IMA 模块中领域服务组件中显示控制的逻辑处理功能的模型。逻辑处理模型是控制显示屏切换的状态机模型,该模型直接从领域服务组件库中提取应用程序。

与传统架构比较,本案例基于开放式航电软件架构进行开发,按照架构规范进行系统软件设计,应用领域服务组件,根据系统需求进行配置,实现了插件式的快速开发部署,降低系统复杂度的同时,极大提高了开发效率;部分功能基于模型和标准化接口开发,可实现功能软件的跨平台移植,大大提升了系统集成的互操作性。

5 结论

本文在实际项目研究工作的基础上,分析了 ASAAC、ARINC 653、POSIX、FACE 等开放标准和 IT 新技术的相关开放技术规范,提出了能力驱动的开放式航电软件架构。该架构主要提供跨平台的软件计算环境;航空电子软件架构规范。在案例研究部分,本文以 ARINC 661 典型显示系统软件开发

为例,阐述了能力驱动的开放式航电软件架构的具体实现,从而证明了能力驱动的开放式航电软件架构的可行性和有效性。

参考文献

- [1] Ronald Koontz, Dale Johnson. Apache mission process software architecture: future airborne capability environment (FACE™) considerations[C]. The AHS 70th Annual Forum, Montreal, Quebec, Canada: the American Helicopter Society International, Inc. 2014.
- [2] 国际系统工程协会(INCose)著,张新国,译.系统工程手册:系统生命周期流程和活动指南[M].北京:机械工业出版社,2017: 614-685.
- [3] Tompkins S. Technical standard for future airborne capability environment (FACE™)[S]. Edition 3.1, The Open Group, 2020.
- [4] J Marques, A Cunha. A reference method for airborne software requirements [C]. 32nd Digital Avionics Systems Conference, 2013.
- [5] ARINC. Specification 661-4, cockpit display system interfaces to user systems [S]. ARINC, 2010.