

航天器软件的研发路线*

吕良庆

(中国科学院国家空间科学中心 复杂航天系统电子信息技术重点实验室, 北京 100190)

摘要:航天器自主能力构建面临着需求不断增多、多变、复杂化、未知的发展趋势,需要在软件研发方式和架构设计上做出改变。在分析已有的4种研发路线的基础上,提出第5条路线,即基于模型化、数据化设计思想的研发路线;提出具有即插即用能力的业务模型,采用电子数据单技术自底向上构建设备级和功能业务级的即插即用能力;以智能规划业务为例说明了业务模型的设计应用。路线5的目标是为构建航天器自主能力和星地之间智能管控能力提供可持续发展的基础。

关键词:软件研发方式;软件架构;即插即用;业务模型;电子数据单

中图分类号:TP311.5;V57 **文献标志码:**A **文章编号:**1001-2486(2021)02-061-05

Research and development method of software on spacecraft

LYU Liangqing

(Key Laboratory of Electronics and Information Technology for Space Systems, National Space Science Center, Chinese Academy of Sciences, Beijing 100190, China)

Abstract: Construction of autonomic capability on spacecraft faces such developing problems as requirements increasing, variability, more complexity, and unknowingness of trend. It is needed to alter development method and architecture design to suit these trends. Based on the research of current four methods, a fifth method, a development mode which is model-based and data-oriented, was proposed. A service model with plug-and-play capability was given, and the electronic data sheet technology was used to construct the bottom-up plug-and-play capability from device level to functional service level. An intelligent planning service instance using this service model was designed and described. The purpose of the fifth method is to provide a persistent development base for the construction of the autonomic capability on spacecraft and the intelligence management and control ability between space and ground.

Keywords: software development mode; software architecture; plug-and-play; service model; electronic data sheet

随着航天器的种类、数量不断增加,任务需求也不断增多且更复杂化、精确化,新的、未知的有效载荷探测设备不断加入,这些设备的工作特点各不相同,并且有可能需要维护、更换、变更任务目标和功能配置,完全依靠地面的管控将会是一项工作量巨大而又复杂的任务,其应对及时性、有效性、精确性都会受到影响。为此建设和提高航天器的智能自主能力是一条势在必行的途径。

支持自主能力的航天器架构和软件系统必须具有控制灵活性和可扩展的特点,在研发过程中就必须加以考虑,并体现在实际运行过程中。

1 航天器软件的研发路线

1.1 路线1:传统研发路线

目前,国内航天软件工程的主流研发路线是

基于结构化设计和瀑布模型的,按照用户需求,针对性地研发软件及其部件。之所以可以这么做,主要是因为航天领域的应用需求具有长时间的稳定性,变化少,各方面质量要求高。它的优势是现行软件工程体系完全能够支持,有其存在的基础和必要性。但是随着航天领域应用业务的不断扩展,这一根基已经在动摇,存在着研发效率低下、不够灵活、成本高和难以重用的问题,难以适应日益复杂、多变的用户任务需求。因此这种路线是需要改造的对象。

1.2 路线2:与地面对等的研发路线

该路线主张航天器上的软件可以使用与地面相同的研发方式,与地面需要的服务和功能有明确的业务对应关系,双方采用对等的协议栈和架构。

* 收稿日期:2019-09-27

基金项目:中国科学院重点部署项目资助项目(ZDRW-KT-2019-5)

作者简介:吕良庆(1969—),男,北京人,研究员,博士,E-mail:lvliangqing@nssc.ac.cn

以空间数据系统咨询委员会 (Consultative Committee for Space Data System, CCSDS) 的任务操作和信息管理业务 (Mission Operations and Information Management Services, MOIMS) 领域^[1-2]为代表的地面研发方式主要是纵向考虑问题的,即按照用户需求,采用数据流分析方法逐层分解,形成相应的架构。这种思路方法也适用于星上软件,可以与路线 1 结合运用。该方法简单易懂,内容明确固定,但适应用户需求变化的能力差,而且星上和地面系统的构成一定是不同的。

硬件方面:受到空间环境的制约,星地的数据处理能力(计算能力、存储能力、网络通信能力)有差距。虽然星上能力也在不断提高,但是需求增长得更快,这种差距会长期存在。

软件方面:地面软件运行时由于可以有人的参与,因此对可靠性和适应未知的要求可以不高,但是在星上无人环境下,需要构建具有智能自主能力的应用层次,以代理地面的(部分)人类智能。

因此,星上软件的架构一定存在与地面架构不同的需要解决的问题和设计,其研发路线也有其特殊性。

1.3 路线 3:模型化研发路线

这种路线期望通过标准化,横向考虑应用支持问题,尽可能提炼公共的部分进行通用化设计,形成标准业务模型(函数库、类库),而不是直接针对用户需求本身。在有具体用户需求时再进行搭积木式的系统构建。这种方式下,用户需求的研发可以基于标准业务模型(如欧洲空间局(European Space Agency, ESA)的包应用标准^[3](Package Utilization Standard, PUS))进行研发、积累和重用,而且特定的用户业务模型也可以转化为标准业务模型。

这种路线适合于长期的标准化模块的研发和积累,但是用户需求的满足以及应用的多样性、未知性仍然留给了具体任务自行解决。

1.4 路线 4:数据化研发路线

在航天器上采用电子数据单(Electronic Data Sheet, EDS)技术,与地面的研发方式、提供的业务没有明确的对应关系,只要能够响应和满足地面所需要的服务即可。

这种路线使用数据描述接口和功能,通过数据配置实现对已有系统的继承、重用,并且其代码自动生成的目标也为星上智能能力的研发、灵活的运控和应对未知问题提供了可能性。它将注意

力放在了数据设计上,需要解决 EDS 的设计和工具链的建设问题,以支持 EDS 的编辑、传递、解析及系统内部的管理信息库^[4-5](Management Information Base, MIB)配置过程。但是,数据化设计需要基于已有的系统架构才能发挥作用、显现出优势,而且对数据的管理是一个需要深入探讨的问题,并不比程序设计简单和容易理解。程序设计变成了辅助性的,考虑的首要问题是通用性,而不是针对性满足用户需求,对研发者提出了更高的要求。

1.5 路线 5:研发路线的综合

上述 4 种路线不是简单的选择问题,而是相互融合的问题,即以路线 1 为基础和出发点,承认其有效性,继承其研发的已有对象和产品,进行重用性改造,形成标准业务,纳入函数库、类库;采用路线 2 的数据流分析设计方法和协议栈,并基于路线 1 的标准业务,进行系统架构的建造和描述;采用路线 3,如 PUS 的方式对标准业务进行归一化设计,并允许标准业务的增加和积累;采用路线 4 的数据化设计思想,进行各种标准业务的解耦合设计,从而增强各种标准业务的通用性和可重用性,适应需求多样化和未知的特点。这种综合可以称为路线 5。

路线 5 的思路是:在归一化的系统架构下,采用标准业务模型,设计用户业务模型;业务模型以 MIB 作为内部数据结构,以 EDS 作为外部数据接口;对 MIB 和 EDS 的数据格式(语法)和内容(语义)进行模型化设计,以支持对业务模型的灵活配置和适应性改造;描述的方式语言化和标准化,采用可扩展标示语言^[6](eXtensive Markup Language, XML)描述,按照 CCSDS 的遥测遥控交换(XML Telemetric and Command Exchange, XTCE)标准^[7-9]和航天器接口业务(Spacecraft Onboard Interface Service, SOIS)的电子数据单^[10-11](SOIS EDS, SEDS)标准,设计相应的编辑器、解析器工具,从而增强系统的自适应性和开发者、用户使用的友好性,融入模型驱动的开发方式(Model-Driven Development, MDD)中。

下面按照这一思路,做一初步探讨。

2 系统架构的构建思路

CCSDS 的 SOIS 领域工作组提出了 SOIS 架构^[12],并针对架构中的各项业务制定了标准建议书。近年来,其研究重心逐渐转移到 EDS 技术方面^[13-14],主张以 SEDS 描述设备访问业务。

按照 SOIS 架构,系统构建需要自底向上逐层

进行。最下层是设备级接入的亚网层^[15],解决各个子网内部即插即用协议设计和设备级 EDS 设计问题。按照 SEDS 标准,采用 XML,形成 SEDS 模板,供工具链设计和接入方使用。在各子网之上构建汇聚层协议,设计归一化的 5 项设备访问业务^[16-20],向上层提供标准化的访问接口,实现系统上层对设备和各子网访问的透明性和接口标准化。而承上启下的数据关系均通过 SEDS 进行描述,以实现业务的即插即用,创造通用的系统构建方法,以适应需求多样、复杂和未知的特点,为以后的智能能力增长和在轨自动编程提供一致的数据基础。

不同机构组织都有自己的体系架构,例如 NASA 的核心飞行软件执行/核心飞行软件或核心飞行软件系统^[21-23] (core Flight software Executive/core Flight Software、core Flight software System, cFE/cFS) 和欧空局的航天航空开放接口架构^[24] (Space AVionics Open Interface aRchitecture, SAVOIR) 等。在已有架构的基础上,按照 SOIS 架构思路,采用 SEDS 技术进行数据化描述,结合 PUS 进行归一化、服务化改造,形成以功能为业务单元、MIB 为单元核心、EDS 为单元间的纽带,可配置、可即插即用,基础夯实,向上开放并可不断扩展的体系架构。在架构的最上层,可以部属不同专业领域的智能算法,以应对不同学科、不同类型的应用需求,而同时又可以继承已有的归一化架构。

在航天器上可以自主闭环管控的基础上,还需要与地面构成闭环管控的关系,从而与传统航天器运控体系相衔接。这种衔接设计有两个要点:一是 SEDS 和 XTCE 标准均采用 XML,具有互通性。XTCE 主要描述遥测遥控数据,可以在地面形成通用遥控指令库和遥测报告库。SEDS 适用于星上设备、业务和应用的描述,对 XTCE 描述的遥测遥控数据可以直接引用,从而构成星地一致的数据规范和实例积累。二是按照 PUS 标准业务模型和用户业务模型进行剪裁、归纳,在星地两端按照各自的研发方式设计实现,为星地和星上智能自主控制两层闭环管控系统提供一致的模型规范。

3 即插即用概念的扩展

即插即用是指当一个设备接入系统时,由系统在运行过程中动态地进行检测和配置的能力^[25-26]。这一概念在航天器上可扩展为不仅是设备的即插即用,也包括了功能业务的即插即用,

表现为可以提供的服务是即插即用的。

3.1 设备即插即用

航天工程中设备即插即用可以用于研发过程中的接口设计、协调和系统自动集成的工作。异构系统是由不同机构组织按照各自的思路设计的,在需要进行互联互通时,异构系统及其软件系统提供的功能业务往往是继承使用,而不是重新设计。为实现异构系统之间的即插即用,采用 EDS 的形式传递相互的配置信息,传递的过程如图 1 所示。

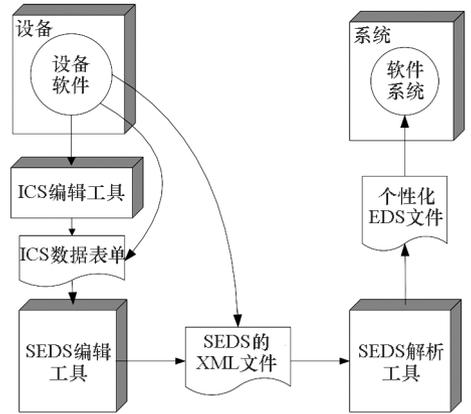


图 1 异构系统之间的 EDS 转换过程^[27]

Fig. 1 EDS conversion process between heterogeneous systems^[27]

图 1 中,设备方可以将自身的信息按照系统要求的实现一致性声明^[10] (Implementation Conformance Statement, ICS) 的规则,使用 ICS 编辑工具生成或手工填写数据表单,再使用 SEDS 编辑工具或手工编辑一次性生成 SEDS 的 XML 文件,从而最大限度地降低了设备方使用 EDS 的难度和工作量,且不改变约定接口数据单 (Interface Data Sheet, IDS) 的工作习惯。系统方则按照 SEDS 标准解析接入方的 XML 文件,形成系统可识别的个性化 EDS 文件。通过这一转换过程,互联双方可以表达自身的需求,了解对方提供的服务能力,达到需求自动匹配的效果,有利于接口标准化以及非标准化设备的继承使用。

个性化 EDS 是系统对外的数据隔离,可以用以继承系统内部已有的、未必规范的数据设计。这是因为系统内部不同业务的 MIB 和使用的 EDS 内容千差万别,且都是针对性的,难以统一。在进行系统新增业务设计时,按照 ICS、SEDS XML 和个性化 EDS 的思路进行规范化设计,可以实现新增部分和继承部分在 EDS 表达方式上的统一,方便系统业务的积累。

图 1 的系统互联互通有三种使用场景:一是

单纯的设备接入系统的场景,设备的 SEDS XML 文件表达的是自描述信息,可以被任何能识别这种信息的系统所接纳。二是互联双方是对等系统,以 SEDS XML 文件为界,各自研发各自的工具。双方各有自己的 ICS 编辑工具,SEDS 编辑和解析工具。三是将这种 EDS 的编辑和解析能力配置在航天器上,就有可能实现两个互不相识的航天器系统在轨飞行过程中的互相自动识别和配置。

3.2 业务即插即用

按照程序设计原理,一个业务主要由程序代码和数据结构两部分组成。要实现功能业务的即插即用,需要建立可识别 EDS 的业务模型,如图 2 所示。

图 2 中,业务的内部数据结构可以统一在 MIB 的概念下,按照架构中的 MIB 和 EDS 组织规则^[5]进行设计,因业务的功能模型不同而不同。其主要思路是 MIB 应有明确的格式,内容可配置;EDS 格式应可动态修改和扩充,以允许同属一个业务模型的不同业务实例的设计,适应不同系统的不同配置需要。业务程序部分除了常规的业务功能及其输入、输出外,为了解决业务可配置和自描述的问题,可基于 MIB 设计相应 EDS 转换功能。如果输入输出的 EDS 与业务 MIB 中的 EDS

项的内容格式是匹配的,则可以省略这种转换,与功能业务的输入输出相统一。

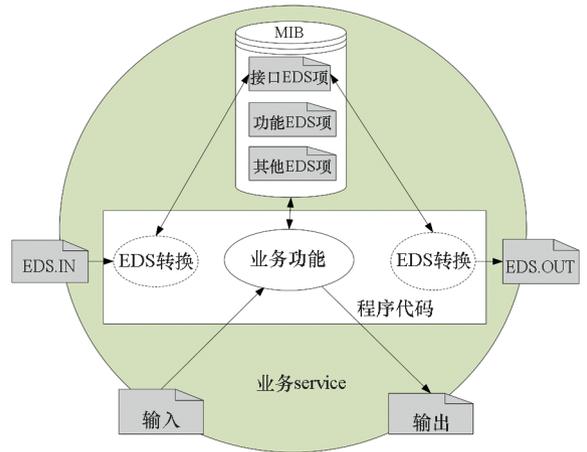


图 2 即插即用业务模型

Fig. 2 Service model with plug-and-play

按照这一业务模型,以智能规划业务为例,其业务模型设计如图 3 所示。

图 3 中,为支持任务规划,需要构建规划知识库,内容包括规划模型库、规划规则库和指令库。这 3 种库可以通过个性化的 EDS 进行配置,如果存在知识表示的不一致,就需要进行配置信息转换。同样,如果提供本业务的知识库描述 EDS 给其他业务,也存在这种转换的需要。

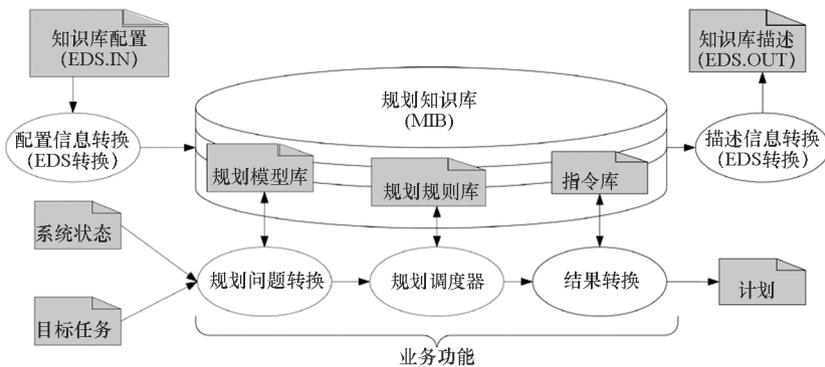


图 3 智能规划业务模型的设计

Fig. 3 Design of intelligent plan service model

在常规的业务功能设计上,规划调度器可以采用现成的调度规划算法(如基于分层任务网络(Hierarchical Task Networks, HTN)算法的 JSHOP2 规划器^[28])进行设计改造。输入的当前系统状态和要达到的任务目标需要根据规划调度器的需要进行转换,转换的依据是规划模型,而转换结果又可以作为新的规划模型保存。规划调度器按照规划规则执行规划算法,输出结果需要根据系统已有的指令配置进行转换,形成具体的可执行计划

输出,从而与具体的系统执行机制相衔接,同时也可以包装成更为高级的指令保存,作为后续更高级规划的基础,表现为自学习能力。

4 结论

路线 5 的研发路线不是单一路线的选择和改进,而是贯彻模型化、数据化设计思想的综合。它的首要目标不是直接满足用户需求,而是系统设计内容是否具有归一化、标准化、可配置、可重用

的特征。基于这一认识所建立的星地闭环系统,其适应用户需求的能力会得到增强,不仅表现在技术指标方面,还表现在研发效率和成本上。在设计上需要解决 EDS 标准的采用、内容格式设计,以及工具链的设计问题,是设备级、功能业务级乃至系统级即插即用能力建设的主要内容。在可持续方面需要走标准化道路,以支持这个方向的研究工作走向工程实用阶段。

参考文献 (References)

- [1] CCSDS. Mission operations services concept: 520.0-G-3[S]. Washington: CCSDS, 2010.
- [2] CCSDS. Mission operations reference model: 520.1-M-1[S]. Washington: CCSDS, 2010.
- [3] ECSS-E-ST-70-41C. Telemetry and telecommand packet utilization[S]. Noordwijk, The Netherlands: ECSS, 2016.
- [4] 杨震, 朱光明. 基于信息管理技术的监控显示系统设计与实现[C]// 第二十届中国空间科学学会空间探测专业委员会学术会议, 2007: 688-691.
YANG Zhen, ZHU Guangming. Design and implementation of monitor and control display system based on information management technology [C]// Proceedings of the 20th Academic Conference of Chinese Society of Space Science Space Exploration Committee, 2007: 688-691. (in Chinese)
- [5] 吕良庆, 黄永辉, 安军社. 基于 CCSDS-SOIS 的航天器数据管理系统体系架构的探讨[J]. 南京大学学报(自然科学), 2018, 54(3): 506-514.
LYU Liangqing, HUANG Yonghui, AN Junshe. Discussion on onboard data management system architecture based on CCSDS-SOIS [J]. Journal of Nanjing University (Natural Science), 2018, 54(3): 506-514. (in Chinese)
- [6] DAVID Hunter, JEFF Rafter, JOE Fawcett, 等. XML 入门经典[M]. 吴文国, 译. 4 版. 北京: 清华大学出版社, 2009.
DAVID Hunter, JEFF Rafter, JOE Fawcett, et al. Beginning XML [M]. Translated by WU Wenguo. 4th ed. Beijing: Tsinghua University Press, 2009. (in Chinese)
- [7] CCSDS. XML telemetric and command exchange (XTCE): 660.0-G-1[S]. Washington: CCSDS, 2006.
- [8] CCSDS. XML telemetric and command exchange (XTCE): 660.0-B-1[S]. Washington: CCSDS, 2007.
- [9] CCSDS. XML formatted data unit (XFDU) structure and construction rules: 661.0-B-1[S]. Washington: CCSDS, 2008.
- [10] CCSDS. Spacecraft onboard interface services—XML specification for electronic data sheets: 876.0-B-1[S]. Washington: CCSDS, 2019.
- [11] CCSDS. Spacecraft onboard interface services—specification for dictionary of terms for electronic data sheets: 876.1-R-2[S]. Washington: CCSDS, 2016.
- [12] CCSDS. Spacecraft onboard interface services: 850.0-G-2[S]. Washington: CCSDS, 2013.
- [13] WILMOT J. Using CCSDS standards to reduce mission costs[C]// Proceedings of 31st Annual AIAA/USU Conference on Small Satellites, 2016.
- [14] MAREK P, JORGE L T, SABINE K. Spacecraft onboard interface services: current status and roadmap [C]// Proceedings of DASIA — Data Systems in Aerospace, 2016.
- [15] 赵和平, 何熊文, 刘崇华, 等. 空间数据系统[M]. 北京: 北京理工大学出版社, 2018.
ZHAO Heping, HE Xiongwen, LIU Chonghua, et al. Space data system [M]. Beijing: Beijing Institute of Technology Press, 2018. (in Chinese)
- [16] CCSDS. Spacecraft onboard interface services—subnetwork packet service: 851.0-M-1[S]. Washington: CCSDS, 2009.
- [17] CCSDS. Spacecraft onboard interface services—subnetwork memory access service: 852.0-M-1[S]. Washington: CCSDS, 2009.
- [18] CCSDS. Spacecraft onboard interface services—subnetwork synchronization service: 853.0-M-1[S]. Washington: CCSDS, 2009.
- [19] CCSDS. Spacecraft onboard interface services—subnetwork device discovery service: 854.0-M-1[S]. Washington: CCSDS, 2009.
- [20] CCSDS. Spacecraft onboard interface services—subnetwork test service: 855.0-M-1[S]. Washington: CCSDS, 2009.
- [21] CUDMORE A, CRUM G, SHEIKH S, et al. Big software for SmallSats: adapting cFS to CubeSat missions [C]// Proceedings of AIAA/USU Conference on Small Satellites, NASA Goddard Space Flight Center, 2015.
- [22] MCCOMAS D. Lessons from 30 years of flight software[R]. MIT Lincoln Labs Software Engineering Symposium, 2015.
- [23] MCCOMAS D, STREGE S, WILMOT J. Core flight system (cFS): a low cost solution for SmallSats [R]. Annual Small Satellite Conference, 2015.
- [24] TERRAILLON J L. SAVOIR: reusing specifications to favour product lines[C]. Toulouse, France: ERTS, 2016.
- [25] 尤政, 田贺祥, 李滨, 等. 微小卫星综合电子系统中的即插即用技术[J]. 清华大学学报(自然科学版), 2009, 49(11): 1765-1769.
YOU Zheng, TIAN Hexiang, LI Bin, et al. Plug-and-play technology for an integration electronic system in a micro-satellite [J]. Journal of Tsinghua University (Science & Technology), 2009, 49(11): 1765-1769. (in Chinese)
- [26] 汪立萍, 蔡亚梅, 陈利玲. 即插即用技术与 PnPSat-1 卫星解析[J]. 空间电子技术, 2013, 10(2): 105-108.
WANG Liping, CAI Yamei, CHEN Liling. Plug-and-play technology and analysis of PnPSat-1 [J]. Space Electronic Technology, 2013, 10(2): 105-108. (in Chinese)
- [27] 张煦冬. EDS 技术在航天数据设计中的应用研究[D]. 北京: 中国科学院大学, 2019.
ZHANG Xudong. Applied study of EDS technology in aerospace data design [D]. Beijing: University of Chinese Academy of Sciences, 2019. (in Chinese)
- [28] 吴迪. 基于 HTN 规划技术在航天器上的应用研究[D]. 北京: 中国科学院大学, 2019.
WU Di. Application research on spacecraft based on HTN planning technology [D]. Beijing: University of Chinese Academy of Sciences, 2019. (in Chinese)