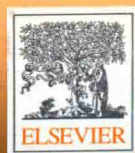
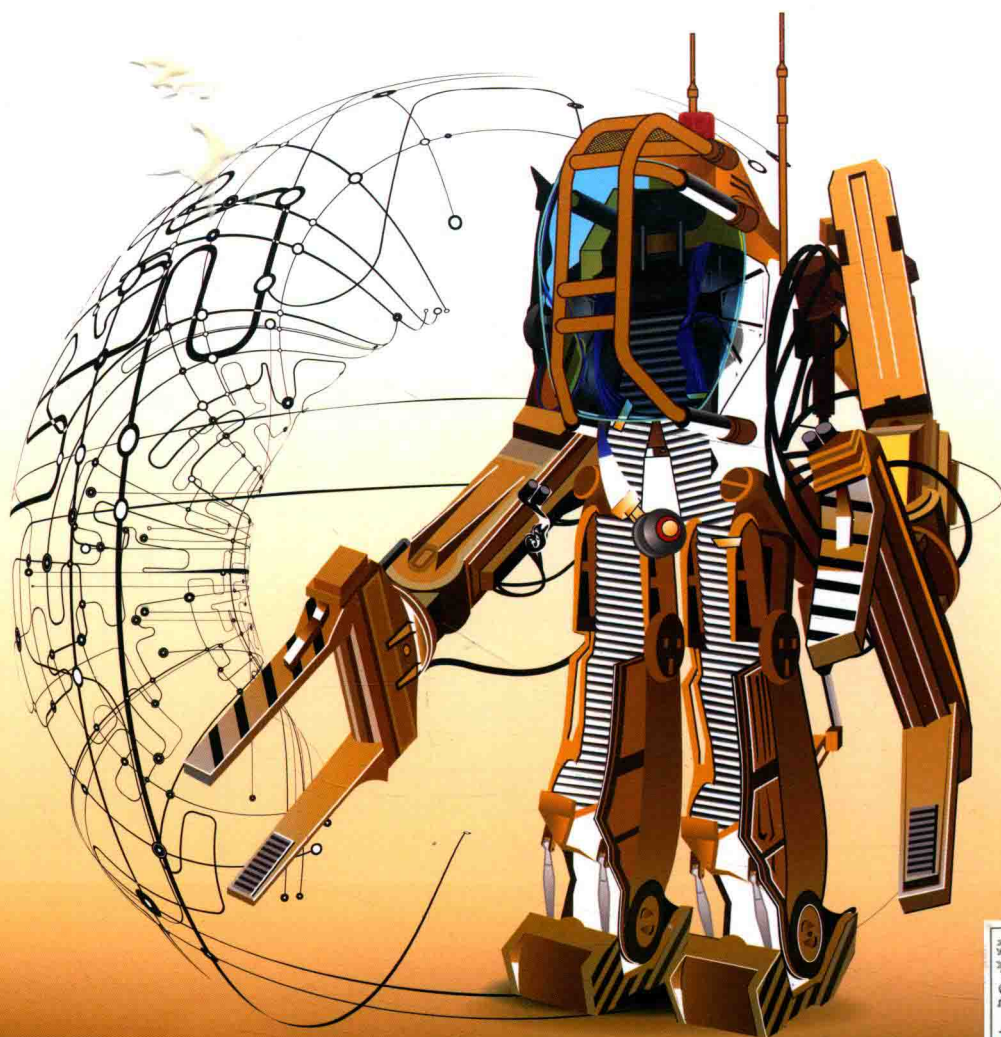


Agile Systems Engineering

敏捷系统工程



[美] Bruce Powel Douglass 著

张新国 谷 炼

译

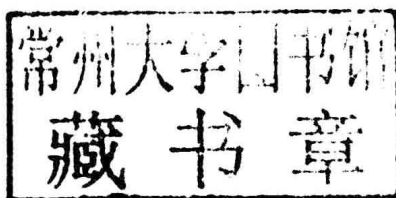
清华大学出版社

ELSEVIER

敏捷系统工程

[美] Bruce Powel Douglass 著

张新国 谷 炼 译



清华大学出版社

北 京

北京市版权局著作权合同登记号 图字：01-2016-8812

本书封面贴有 Elsevier 防伪标签，无标签者不得销售。

版权所有，侵权必究。侵权举报电话：010-62782989 13701121933

图书在版编目(CIP)数据

敏捷系统工程 / (美)布鲁斯·鲍威尔·道格拉斯(Bruce Powel Douglass) 著；张新国，谷炼 译. —北京：清华大学出版社，2018 (2018.6重印)

书名原文：Agile Systems Engineering

ISBN 978-7-302-49092-0

I. ①敏… II. ①布… ②张… ③谷… III. ①软件开发 IV. ①TP311.52

中国版本图书馆 CIP 数据核字(2017)第 300294 号

责任编辑：王 军 于 平

封面设计：牛艳敏

版式设计：方加青

责任校对：曹 阳

责任印制：刘海龙

出版发行：清华大学出版社

网 址：<http://www.tup.com.cn>, <http://www.wqbook.com>

地 址：北京清华大学学研大厦 A 座 邮 编：100084

社 总 机：010-62770175 邮 购：010-62786544

投稿与读者服务：010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈：010-62772015, zhiliang@tup.tsinghua.edu.cn

印 装 者：北京密云胶印厂

经 销：全国新华书店

开 本：185mm×260mm 印 张：20.75 字 数：531 千字

版 次：2018 年 2 月第 1 版 印 次：2018 年 6 月第 2 次印刷

定 价：98.00 元

产品编号：069148-01

Elsevier (Singapore) Pte Ltd.
3 Killiney Road, #08-01 Winsland House I, Singapore 239519
Tel: (65) 6349-0200; Fax: (65) 6733-1817

Agile Systems Engineering
Bruce Powel Douglass
© 2016 Elsevier Inc. All rights reserved.
ISBN-13: 9780128021200

This translation of Agile Systems Engineering by Bruce Powel Douglass was undertaken by Tsinghua University Press and is published by arrangement with Elsevier (Singapore) Pte Ltd.

Agile Systems Engineering by Bruce Powel Douglass 由清华大学出版社进行翻译, 并根据清华大学出版社与爱思唯尔(新加坡)私人有限公司的协议约定出版。

《敏捷系统工程》(张新国 谷炼译)

ISBN: 9787302490920

Copyright © 2017 by Elsevier (Singapore) Pte Ltd.

All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system, without permission in writing from Elsevier (Singapore) Pte Ltd. Details on how to seek permission, further information about the Elsevier's permissions policies and arrangements with organizations such as the Copyright Clearance Center and the Copyright Licensing Agency, can be found at our website: www.elsevier.com/permissions.

This book and the individual contributions contained in it are protected under copyright by Elsevier (Singapore) Pte Ltd. and Tsinghua University Press (other than as may be noted herein).

This edition is printed in China by Tsinghua University Press under special arrangement with Elsevier (Singapore) Pte Ltd. This edition is authorized for sale in the People's Republic of China only, excluding Hong Kong SAR, Macau SAR and Taiwan. Unauthorized export of this edition is a violation of the contract.

本书简体中文版由 Elsevier(Singapore) Pte Ltd. 授权 Tsinghua University Press 在中国大陆地区(不包括香港、澳门特别行政区以及台湾地区)出版与发行。未经许可之出口, 视为违反著作权法, 将受民事和刑事法律之制裁。

本书封底贴有 Elsevier 防伪标签, 无标签者不得销售。

注意

本书涉及领域的知识和实践标准在不断变化。新的研究和经验拓展我们的理解, 因此须对研究方法、专业实践或医疗方法作出调整。从业者和研究人员必须始终依靠自身经验和知识来评估和使用本书中提到的所有信息、方法、化合物或本书中描述的实验。在使用这些信息或方法时, 他们应注意自身和他人的安全, 包括注意他们负有专业责任的当事人的安全。在法律允许的最大范围内, 爱思唯尔、译文的原文作者、原文编辑及原文内容提供者均不对因产品责任、疏忽或其他人身或财产伤害及/或损失承担责任, 亦不对由于使用或操作文中提到的方法、产品、说明或思想而导致的人身或财产伤害及/或损失承担责任。

致给予我支持的挚爱夫人和我杰出的

铁人三项引路人

Sarah——你真了不起

译者序

复杂系统的构想、开发、实施和运行带来了技术和管理上的双重挑战，应对系统工程的复杂性已经成为未来工业领域的核心竞争力之一。目前，架构引领、基于模型、数据驱动已经成为解决复杂问题的基本能力要素。面对复杂系统生命周期中呈现出的不可预见性与不确定性、外部背景环境剧烈变化以及新兴技术不断涌现，复杂系统的采办方提出了进化式、渐进式的策略，利益攸关者的需要和需求以及系统需求将不断变化，这就要求复杂系统的开发过程和系统架构应具备良好的敏捷性，即在不可预见性、不确定性和变化的态势下保持系统开发有效开展的能力。

架构引领、基于模型、数据驱动的新的开发模式为敏捷方法在复杂系统工程中的实践和应用提供了基础，使得复杂系统在开发过程中以增量式、迭代式的方式开展。这种新的开发模式被称为基于模型的敏捷系统工程，即aMBSE(agile Model Based Systems Engineering)。aMBSE方法由本书作者Bruce Powel Douglass博士在MBSE方法上首创，广泛应用于复杂系统的开发之中，并被证明是行之有效的。aMBSE方法从复杂系统的某一个或几个方面的需求入手建立模型，并对建立的模型进行执行和分析，在系统开发的早期就进行相关需求和设计的验证和确认，从而避免大量的工作和人力被投入之后产生的风险和费用。

架构方法是解决复杂性问题的有效方法，通过架构方法能够基于彼此逻辑相关且协调一致的原则、概念和特性来创造一个全面的解决方案，进而满足系统需求集合(可追溯到业务或使命以及利益攸关者需求)及生命周期概念(如实施、运行、支持)所表达的问题或机会，并且可通过技术(如机械、电子、软件等)来实现。本书提出并定义了子系统/组件、可依赖性、分布、部署、并发与资源5个视角，进而开发了候选架构的模型和视图，通过对复杂系统从需求到功能架构、逻辑架构再到物理架构的连续传递和映射，实现从系统需求到架构实体再到系统元素的划分、对准和分配，并建立系统设计与演进的指导原则。

传统的基于文本的系统工程已经难以满足复杂系统的开发要求，必须采用正规化的系统建模语言对系统的需求、架构和设计进行表达，保证整个开发过程的正确性、一致性和完整性。本书使用国际面向对象组织OMG制定的正规形式化图形建模语言SysML，对系统的需求、功能、行为、结构进行建模。SysML遵循MOF的四层元模型框架，具备对“功能-行为-结构”逻辑关系的描述能力。功能表达了系统为满足使命将输入转化为输出的动作以及相应的性能，行为表达了系统功能之间的交互关系以及系统状态随时间发生的转移，结构则表达了如何实现这些功能和行为。本书通过使用SysML语言以及图形化的建模环境，从不同维度展现复杂系统的需求、功能、行为和结构。

本书中的另一个亮点是如何对复杂系统中的各种数据进行正确的表述，这是复杂系统工程人员较为棘手的问题。数据是物理世界在赛博空间的映射，可以表述系统与背景环境之间以及系统元素之间物质、能量和信息的交互过程。数据驱动的复杂系统定义必须按照由抽象到具体、由逻辑到物理的逐步细化过程。复杂系统首先被看作一个“黑盒”，识别并定义其与外部交互的逻辑数据和接口。然后对系统架构的备选方案进行权衡和分析，选择架构中不

同组件的实现方式，进而定义不同组件、不同学科之间的物理数据和接口，为基于模型的向下游工程转交提供输入。本书提出了数据模式的概念，定义了复杂系统的逻辑数据模式和物理数据模式，实现了机械、电子、软件等不同领域学科之间的互操作。正是由于系统模型的建立和数据模式的定义，使得敏捷方法从软件领域被推广到多学科领域，引发了系统工程应用模式的转型。

本书作者Bruce Powel Douglass博士是系统工程与嵌入式软件开发领域的知名专家，在该领域拥有超过30年的工作经验。作为嵌入式系统开发的代表性人物，Bruce在结构化和面向对象方法、嵌入式系统开发、安全关键系统设计和复杂系统设计等方面都有很深的造诣。本书以承载能力达1500千克的可穿戴式工业外骨骼机器人以及为人熟知的高端跑步机为示例，使有大量实例阐明在应用aMBSE方法中所涉及的工程流程步骤，使读者更容易掌握aMBSE的精髓。因此，本书自出版之日起已成为Amazon网站上最畅销的科技书籍之一。

综上所述，aMBSE是一种应对复杂系统的有效方法，它实现了从文本到模型、从预估到敏捷、从软件到多学科领域的转变。通过本书的学习，可以为系统工程师提供详细的指导，而使读者能够容易且有效地将aMBSE方法应用到复杂产品开发中。

最后，感谢中国航空工业集团公司信息技术中心高星海、鄢永军、常创业以及编译工作小组在校对、版本控制等方面中付出的努力，在此表示衷心的感谢。

译者

作者简介

Bruce Powel Douglass 3岁时开始自学读书，不到12岁就开始学习微积分。他14岁辍学游历美国，几年后进入俄勒冈大学学习数学专业。他最终获得俄勒冈大学运动生理学科学硕士学位以及USD医学院神经生理学博士学位。他在USD医学院期间提出了一个名为自相关因子分析的数学分支，用于研究多细胞生物神经系统中的信息处理。

Bruce作为软件开发人员和系统工程师在实时嵌入式系统领域已经工作超过30年，是实时嵌入式系统领域著名的演说家、作者与顾问。他是**嵌入式系统大会**和**UML世界大会**的顾问委员会的成员之一，并在会议上讲授过关于系统工程、项目估算和调度、项目管理、面向对象的分析与设计、通信协议、有限状态机、设计模式以及安全性关键系统设计方面的课程。Bruce在实时系统、软件设计以及项目管理方面有多年的开发、授课与咨询经验。他还为很多(特别是实时领域内的)杂志和期刊撰文。

Bruce是IBM物联网(IoT)业务部的首席布道师。作为首席布道师，除了披荆斩棘开拓道路，他更像是一位首席科学家。Bruce与UML合作伙伴在UML与SysML标准的规定方面密切合作。他开发了用于Rhapsody建模工具的第一个DoDAF的UML概要以及其他概要，例如故障树分析概要以及安保性分析概要。他是对象管理组组织的实时分析和设计工作组的副主席。他还撰写了其他几本关于系统与软件开发方面的书籍，包括*Doing Hard Time: Developing Real-Time Systems with UML, Objects, Frameworks and Patterns*(Addison-Wesley, 1999)、*Real-Time Design Patterns: Robust Scalable Architecture for Real-Time Systems*(Addison-Wesley, 2002)、*Real-Time UML 3rd Edition: Advances in the UML for Real-Time Systems*(Addison-Wesley, 2004)、*Real-Time Agility*(Addison-Wesley, 2009)、*Design Patterns for Embedded Systems in C*(Elsevier, 2011)、*Real-Time UML Workshop for Embedded Systems*(Elsevier, 2014)等，以及一本关于乒乓球方面的短篇教材。

Bruce喜欢古典音乐，古典吉他弹奏水平达到专业水准。他参加过多场体育比赛，包括乒乓球、自行车极限马拉松赛、赛跑以及全接触跆拳道，尽管目前还只是与打不还手的静物交手。他最近重新回到三项全能运动比赛以及自行车极限马拉松赛，并在2014年首次参加了铁人三项比赛。

Bruce在全世界进行广泛咨询与培训活动。如果你对此感兴趣，可以通过Bruce.Douglass@us.ibm.com与他联系。

译者简介

张新国，工学博士，管理学博士，研究员，博士生导师。现任中国航空工业集团公司副总经理/CIO，中国航空研究院院长。兼任国际系统工程协会(INCOSE)北京(中国)分会会长，国际航空科学理事会(ICAS)执委会委员，中国航空学会副理事长，国家制造业创新中心专家组成员。英国皇家航空学会(RAeS) Fellow，美国航空航天学会(AIAA) Associate Fellow。清华大学特聘教授，香港科技大学(HKUST)工业咨询委员会顾问。

在复杂系统工程的理论、方法和应用等方面有深入研究，并在航空工业以架构为中心的基于模型的系统工程的工业系统转型升级中不断取得新的创新和突破，是信息化与工业化深度融合在航空工业全面推进的主要领导者之一。

曾荣获“国家留学回国人员成就奖”“全国先进工作者”“全国五一劳动奖章”，并荣获“国家科技进步特等奖”，是享受国务院政府特殊津贴专家。

有多篇论文在国内外重点刊物上发表，并著有《电传飞行控制系统》(2002年，国防工业出版社)、《国防装备系统工程中的成熟度理论与应用》(2013年，国防工业出版社)、《新科学管理——面向复杂性的现代管理理论与方法》(2013年，机械工业出版社)，译著有《系统工程手册——系统生命周期流程和活动指南3.2.2版》(2013年，机械工业出版社)、《基于模型的系统工程(MBSE)方法论综述》(2014年，机械工业出版社)、《TOGAF标准9.1版》(2017年，机械工业出版社)、《系统工程手册——系统生命周期流程和活动指南4.0版》(2017年，机械工业出版社)等书。

谷炼，现就职于德国INCHRON GmbH(www.inchron.com)公司，任亚太区总裁。她目前在带领团队，在亚太区推进INCHRON的解决方案，帮助客户克服在系统工程和软件工程中遇到的时间和性能方面的困难和挑战。

谷炼在系统工程和软件工程领域有着丰富和扎实的经验 and 咨询能力。她曾参与和指导过多个行业的解决方案的设计和 implementation 工作。所涉及的行业包括航天航空及防御领域、汽车及机动车制造、电子工业、电信运营和制造、能源与利用、金融及保险等。

谷炼还曾就职于IBM公司，任亚太区解决方案经理；Telelogic公司，任SCM亚太区产品经理；Rational Software公司，任中国及日本地区的的解决方案高级咨询师。

谷炼毕业于北京科技大学，获工学硕士。之后，获得日本富士通公司的奖学金并在日本再次获得计算机科学硕士。谷炼精通三国语言：中文、英语和日语。多种语言能力给了她跨区域支持和管理业务的能力，并可与不同国家和地区的团队人员密切合作。

前言

产品的功能和复杂性正在成倍地增加,而且对这些系统的安全性、可靠性以及安保性的关注使得这样的系统对工程师而言更加困难。同时,产品开发周期正在萎缩。很显然,变革是需要的。我们需要能够以更少的时间制造出更有能力且缺陷更少的系统。

针对此问题,一个受到高度评价的解决方案是避免以文本作为捕获工程数据的主要手段。虽然文本具有极好的表现力,但是它是有歧义的,而且是极其不严谨的。使用更加正规的定义语言(这里,显然是指UML和SysML)进行建模是要力求改善特定的工程数据。只要我们能够想出改进的方式即可。

另一个所提供的解决方案是敏捷方法。尽管敏捷方法已经开始应用于嵌入式和实时系统,但这些方法却是由软件IT行业开发的。然而,敏捷文献(几乎)完全关注在台式机或IT软件开发上。他们考虑的开发环境(几乎)全部都是同地域小型团队的合作,并不关注安全性、可靠性或安保性问题;而且没有与电子或机械部件的联合开发。因此,系统工程师想要知道的是“这种方法如何适用于‘我’和我的工作”。敏捷文献没有给出答案。

有一些关于系统工程的很好的书籍,也有一些关于SysML与基于模型的系统工程(MBSE)的很好的书籍。有许多关于软件的敏捷方法的书籍(其中一些书籍也是很好的)。然而,目前还没有书籍来尝试将这些概念综合为一种一致且可用的系统工程方法。本书的目的就在于满足这种需要。

我们首先简单地介绍了系统工程学科,之后又简短讨论了敏捷方法,因为它们在大多数系统文献中都有论述,包括其益处。除前言部分外,还有一章内容关于基本的SysML。接着,我们就开始理解如何在现实生活中应用MBSE。

本书中的方法基于作者的Harmony敏捷系统工程流程。该流程有关软件开发方面的部分在其他文献中有详细描述^①;本书仅涉及系统工程的关注点。Harmony敏捷系统工程流程是一种敏捷的、以模型为中心的实施途径,用于开发系统工程所需的工程数据;需求、架构、接口以及可依赖性分析是其中最重要的内容。Harmony流程是依据作者在全球范围内所指导完成、取得飞速进展并在其他方面发挥作用的实际项目上累积的数十年系统经验提出和完善的。

在教育工作者中有这样一种说法——“我示你看。我讲你听。你做你懂”。为此,本书中有大量示例用于阐明执行所涉及的工程步骤的细节。这些示例涉及工程学科的多个方面,包括软件、电子和机械工程。这些示例中的第一个示例是高端跑步机。第二个更复杂的示例是能够承载1500千克的可穿戴工业用机器人外骨骼(被称为waldo)。Harmony敏捷系统工程流程的每个主要活动都是以这些和其他示例展开讨论和演示的。我们鼓励读者针对提出的问题构建自己的解决方案并建立这些章节中所描述的模型。

^① 例如,参见*Real-Time Agility* (Addison-Wesley, 2009)或*Real-Time UML Workshop for Embedded Systems* (Elsevier, 2014)。

读者

本书的主要读者不言而喻是系统工程师。系统工程师的主要关注点集中在(通常)由多个工程学科所实施的系统规范与设计上。系统工程师规定了产品的系统特性而将学科特有的细节留给适当的下游工程团队。一些下游工程师也可能在本书中找到感兴趣的信息,特别是系统工程数据如何被格式化并采纳以满足转交活动中他们需要的细节。

目标

在游历世界期间,我感受到系统工程师在应用MBSE方法时所遇到的困难。主要的语言——SysML——令人望而生畏。SysML包括800页左右的UML规范并且增加了数百页。它是一种功能极为强大但是十分复杂的语言。

除了语言本身,随着产品复杂性成倍增加以及产品交付周期的不断缩减,亟须同时提高系统工程工作的效率并改进质量。我们看到系统在安全关键的、高可靠性和安保性环境中正日益取代人类,并且我们必须能够始终依靠这些系统的功能正常运转。

本书有一个简单目标——为系统工程师提供足够的指导,以便他们能方便有效地将敏捷方法和MBSE应用到复杂系统的开发中,因为现实世界越来越依赖于这些系统的运行。

工具

本书中的所有建模示例都使用IBM® Rhapsody™工具进行建模。然而,关于标准的一个好处是对不同工具的多种选择。如果你偏爱的其他工具支持SysML标准,那么你用你选择的工具建立这些模型时应该不会遇到什么困难。这不是一本关于Rhapsody的书,也不是专用于Rhapsody的书。

拓展

如果你对工具、培训或咨询感兴趣,参见www.ibm.com。我在世界范围内教授关于UML、SysML、MDA、DoDAF、架构设计、设计模式、需求建模、用例、安全性关键开发、行为建模、开发流程改进、项目管理与调度等多门高等课程并提供咨询。你可通过Bruce.Douglass@us.ibm.com就培训或咨询服务与我联系。我还开通了一个(免费的)yahoo群组论坛,网址是<http://groups.yahoo.com/group/RT-UML>——快来参与吧! My IBM Thought Leader页面(<http://www-01.ibm.com/software/rational/leadership/thought/brucedouglass.html>)也包含你可能感兴趣的白皮书,其涉及不同课题并可供下载。

Bruce Powel Douglass博士

致 谢

我要感谢我的编辑Charlotte Kent恰到好处的挑剔，并在需要帮助时所给予的支持。我的审稿人(IBM的Barclay Brown)也帮助我保持坦诚。毋庸置疑，文中仍存在错误——我对此承担全部责任。最重要的是，我要向我的家人表示感谢，他们在我挑灯熬夜写书期间设法让我时刻保持头脑清醒。

目 录

第1章 什么是基于模型的系统工程... 1

1.1 关键的系统工程活动..... 1	
1.1.1 识别客户需要..... 2	
1.1.2 规定系统需求..... 2	
1.1.3 评估可依赖性..... 3	
1.1.4 评价备选架构和技术..... 3	
1.1.5 选择特定架构和技术..... 4	
1.1.6 分配需求和接口到架构..... 4	
1.1.7 向下游工程转交..... 4	
1.1.8 将学科特定的设计综合至系统 组成..... 5	
1.1.9 以整体验证系统..... 5	
1.1.10 系统确认..... 8	
1.2 系统工程数据..... 8	
1.2.1 系统开发规划..... 8	
1.2.2 利益攸关者需求..... 9	
1.2.3 系统需求..... 9	
1.2.4 认证规划..... 9	
1.2.5 子系统需求..... 9	
1.2.6 学科特定的需求..... 9	
1.2.7 安全性分析..... 10	
1.2.8 可靠性分析..... 10	
1.2.9 安保性分析..... 10	
1.2.10 系统架构..... 10	
1.2.11 综合测试规划..... 11	
1.2.12 综合测试..... 11	
1.2.13 验证规划..... 11	
1.2.14 验证试验..... 12	
1.2.15 确认规划..... 12	
1.2.16 追溯矩阵..... 12	
1.2.17 综合测试结果..... 13	
1.2.18 验证结果..... 13	
1.2.19 确认结果..... 13	

1.3 系统工程的生命周期..... 13	
1.3.1 V模型生命周期..... 13	
1.3.2 增量式..... 15	
1.3.3 混合式..... 16	
1.4 基于模型的系统工程(MBSE)..... 17	
1.4.1 建模的优势..... 17	
1.4.2 用UML和SysML进行高精度 建模..... 20	
1.4.3 建模是敏捷系统工程的根本..... 20	
1.4.4 在你的组织或项目中采纳 建模..... 21	
1.4.5 建模规则..... 25	
1.5 总结..... 27	
参考文献..... 27	

第2章 什么是敏捷方法..... 29

2.1 敏捷宣言..... 30	
2.2 敏捷方法的益处..... 32	
2.2.1 提高工程数据的质量..... 32	
2.2.2 提高工程效率..... 32	
2.2.3 尽早获得投资的回报(ROI)..... 33	
2.2.4 利益攸关者满意..... 33	
2.2.5 增强了项目控制..... 33	
2.2.6 响应变化..... 33	
2.2.7 更早且更大幅度地降低项目 风险..... 33	
2.3 将敏捷宣言应用于系统工程..... 34	
2.3.1 增量式地工作..... 34	
2.3.2 动态地规划..... 34	
2.3.3 主动降低项目风险..... 35	
2.3.4 持续地验证..... 36	
2.3.5 连续地综合..... 36	
2.3.6 用例1: 在空域中发现轨迹..... 36	

2.3.7 用例2：进行定期的内置测试(PBIT)..... 36

2.3.8 频繁地确认..... 37

2.3.9 建模是aMBSE的根本..... 37

2.4 针对系统工程的敏捷最佳实践..... 37

2.4.1 工作产品的增量式开发..... 38

2.4.2 工作产品的持续验证..... 38

2.4.3 可执行的需求模型..... 39

2.4.4 链接到文本规范的基于模型的规范..... 41

2.4.5 连续的可依赖性评估..... 41

2.4.6 主动的项目风险管理..... 42

2.4.7 向下游工程的基于模型的转交... 43

2.4.8 动态的规划..... 43

2.5 汇总：Harmony aMBSE流程..... 45

2.5.1 启动项目..... 47

2.5.2 定义利益攸关者需求..... 49

2.5.3 系统需求定义和分析..... 50

2.5.4 途径1：基于流的用例分析..... 51

2.5.5 途径2：基于场景的用例分析... 51

2.5.6 途径3：基于状态的用例分析... 52

2.5.7 架构分析..... 53

2.5.8 架构设计..... 55

2.5.9 进行迭代回顾..... 56

2.5.10 向下游工程转交..... 57

2.5.11 控制项目..... 58

2.5.12 进行品质保证审计..... 59

2.5.13 管理变更..... 59

2.6 总结..... 59

参考文献..... 60

第3章 SysML介绍.....61

3.1 SysML概览..... 61

3.2 UML扩展机制..... 64

3.2.1 SysML模型元素..... 65

3.2.2 SysML图..... 66

3.2.3 行为图..... 67

3.2.4 需求图..... 68

3.2.5 结构图..... 69

3.3 组织你的模型很重要..... 72

3.4 关键SysML视图和核心语义..... 76

3.4.1 块、关系、接口和端口..... 76

3.4.2 顺序图..... 86

3.4.3 活动、动作和活动图..... 89

3.4.4 状态机图..... 94

3.5 最小SysML概要..... 103

3.6 总结..... 105

3.6.1 摘自UML..... 105

3.6.2 修改..... 105

3.6.3 新元素..... 106

参考文献..... 106

第4章 敏捷的利益攸关者需求工程.....107

4.1 目标..... 107

4.2 利益攸关者需求 workflow..... 107

4.2.1 牢记——这是敏捷MBSE..... 109

4.2.2 什么是用例..... 109

4.2.3 用例图..... 112

4.3 示例模型：T-Wrecks工业外骨骼... 116

4.4 识别利益攸关者..... 117

4.4.1 驾驶员..... 118

4.4.2 机队管理人员..... 118

4.4.3 维护人员..... 118

4.4.4 采购方..... 118

4.4.5 安装人员..... 119

4.4.6 T-Wreckers测试团队..... 119

4.4.7 制造工程师..... 119

4.5 生成利益攸关者需求..... 119

4.5.1 什么是需求..... 119

4.5.2 性能需求和其他QoS需求..... 121

4.5.3 需求可视化..... 122

4.5.4 需求管理工具..... 124

4.5.5 组织利益攸关者需求规范..... 124

4.6 对利益攸关者用例场景建模..... 124

4.6.1 什么是用例场景..... 125

4.6.2 场景分析 workflow..... 127

4.6.3 T-Wrecks利益攸关者用例场景..... 129

4.7 创建/更新确认规划	135	6.2.2 定义备选解决方案	209
4.8 总结	136	6.2.3 架构权衡研究	209
4.8.1 识别利益攸关者	136	6.2.4 将多个解决方案并入系统 架构	210
4.8.2 生成利益攸关者需求	136	6.2.5 定义评估准则	210
4.8.3 对利益攸关者用例场景建模	136	6.2.6 向准则分配权重	210
4.8.4 创建/更新确认规划	137	6.2.7 为每个准则定义效用曲线	211
4.9 未完待续	137	6.2.8 向众多备选解决方案分配 MOE	211
参考文献	137	6.2.9 确定解决方案	211
第5章 敏捷的系统需求定义和 分析	139	6.3 评估方法	211
5.1 目标	139	6.3.1 简单方法	211
5.2 系统需求工作流	139	6.3.2 高保真方法	213
5.2.1 识别系统用例	140	6.4 识别关键的系统功能(和特性)	216
5.2.2 生成/更新系统需求	141	6.5 定义备选解决方案	218
5.2.3 进行用例分析	141	6.5.1 Speed Demon备选解决方案	218
5.2.4 创建逻辑数据模式	142	6.5.2 T-Wrecks备选解决方案	219
5.2.5 分析可依赖性	142	6.6 进行架构权衡研究	222
5.2.6 创建/更新系统验证规划	142	6.6.1 定义评估准则	222
5.3 识别系统用例	142	6.6.2 向准则分配权重	223
5.4 生成系统需求	143	6.6.3 为每个准则定义效用曲线	224
5.5 分析用例	144	6.6.4 向备选解决方案分配MOE	226
5.5.1 基于流的用例分析	144	6.6.5 确定解决方案	229
5.5.2 基于场景的用例分析	160	6.7 将多个解决方案并入系统架构	229
5.5.3 基于状态的用例分析	176	6.8 总结	230
5.6 创建/更新逻辑数据模式	189	6.9 未完待续	230
5.7 可依赖性分析	192	参考文献	230
5.7.1 安全性分析	192	第7章 敏捷的系统架构设计	231
5.7.2 T-Wrecks初始可依赖性分析	201	7.1 目标	231
5.8 创建/更新验证规划	204	7.1.1 什么是子系统	231
5.9 总结	204	7.1.2 关键架构视图	232
5.10 未完待续	205	7.2 架构设计工作流	234
参考文献	205	7.2.1 识别子系统	234
第6章 敏捷的系统架构分析与权衡 研究	207	7.2.2 向子系统分配系统需求	234
6.1 目标	207	7.2.3 向子系统分配用例	235
6.2 架构分析工作流	208	7.2.4 创建/更新逻辑数据模式	235
6.2.1 识别关键的系统功能	209	7.2.5 创建/更新子系统需求	235
		7.2.6 开发控制律	235

7.2.7 分析可依赖性.....	235	8.2.1 收集子系统规范数据.....	275
7.2.8 进行评审.....	236	8.2.2 创建共享模型.....	276
7.3 识别子系统.....	236	8.2.3 定义子系统物理接口.....	276
7.3.1 Speed Demon子系统.....	237	8.2.4 创建子系统模型.....	277
7.3.2 T-Wrecks子系统.....	245	8.2.5 定义跨学科接口.....	277
7.4 向子系统分配系统需求.....	248	8.2.6 将需求分配到工程学科.....	277
7.5 向子系统分配用例.....	249	8.3 收集子系统规范数据.....	277
7.5.1 自底向上分配.....	250	8.3.1 收集SysML模型数据.....	277
7.5.2 自顶向下分配.....	251	8.3.2 收集其他工程数据.....	278
7.5.3 公共任务.....	253	8.4 创建共享模型.....	279
7.5.4 Speed Demon子系统用例分配 示例.....	254	8.5 定义子系统物理接口.....	280
7.5.5 T-Wrecks子系统用例分配 示例.....	259	8.5.1 从逻辑规范中创建物理规范.....	281
7.6 创建/更新逻辑数据模式.....	265	8.5.2 Speed Demon接口示例.....	284
7.6.1 Speed Demon跑步机示例.....	266	8.5.3 T-Wrecks接口示例.....	287
7.6.2 T-Wrecks示例.....	267	8.6 创建子系统模型.....	290
7.7 创建/更新子系统需求.....	268	8.7 定义跨学科接口.....	291
7.8 开发控制律.....	269	8.7.1 Speed Demon示例: Control Unit 子系统接口规范.....	291
7.9 分析可依赖性.....	270	8.7.2 T-Wrecks示例.....	293
7.9.1 安全性分析.....	271	8.8 将需求分配到工程学科.....	297
7.9.2 可靠性分析.....	271	8.8.1 Speed Demon示例.....	298
7.9.3 安保性分析.....	271	8.8.2 T-Wrecks示例.....	299
7.10 总结.....	271	8.9 下游工程开始.....	304
7.11 未完待续.....	272	8.10 系统工程还在继续.....	305
参考文献.....	272	参考文献.....	305
第8章 向下游工程转交.....	275	附录A T-Wrecks利益攸关者需求 ...	307
8.1 目标.....	275	附录B T-Wrecks系统需求	311
8.2 向下游工程转交的工作流.....	275		

什么是基于模型的系统工程

系统工程是跨学科的活动，它更多地关注系统特性而不是特定的技术，其总体目标是产生优化的系统以满足潜在的复杂需要。该关注包括必要的系统特性规范(需求)、大规模系统的组织原则(系统架构)、在其环境中系统和元素间以及构成该系统的大型架构元素间移动的“流”和事件的定义(接口)以及通过优化分析对关键方法和技术的选择(权衡研究)。

简而言之，系统工程是构建复杂的技术多元化系统的跨学科方法。

贯穿于系统规范、开发和验证的活动都涉及系统工程。系统工程提供关键的系统级项目监管。在本书中，我们将关注规范活动，但系统工程所包括的远不止这些。

系统工程区别于学科特定工程，学科特定工程统称为“下游工程”。这些工程学科包括机械、电子、化学、光学、核电和软件工程。因此，当系统工程可以定义分配到这些特定工程学科的需求时，这些需求不应规定这些学科中所采用的设计或技术，除非是在高的层级上。

1.1 关键的系统工程活动

图1.1指明系统工程的主要方面，它们连接成“传统的”或“经典的”流。

当然，系统工程比这要多很多，在本书中，我们会涵盖很多方面。当前，它提供一套有用的高层次的论点。关于更多的正式的系统工程定义，请读者参考 *INCOSE Systems Engineering Handbook* [1]。

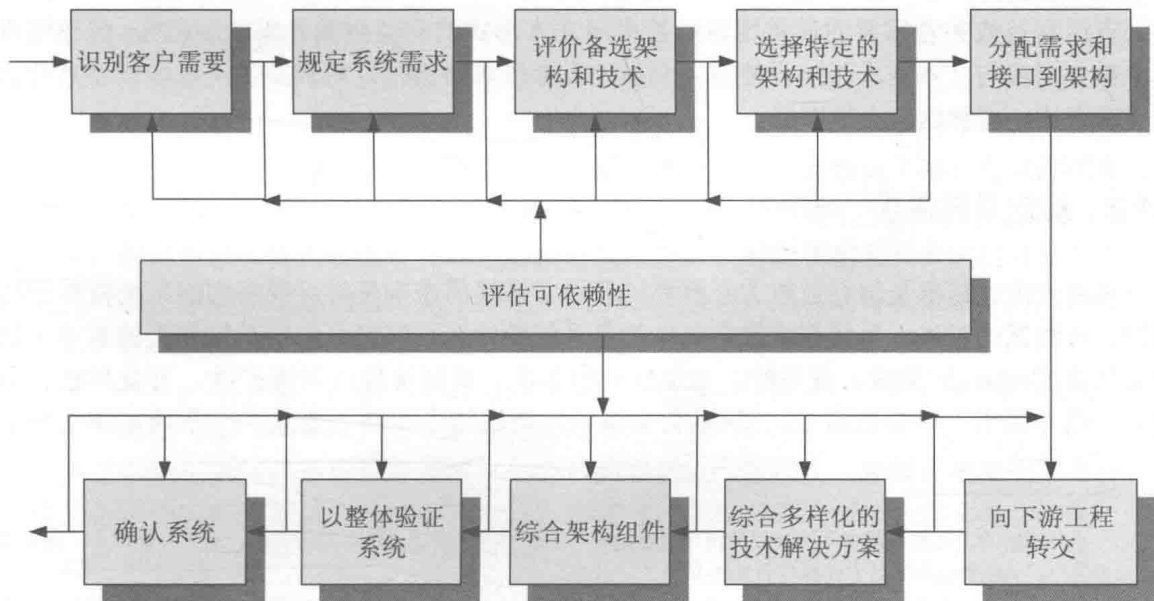


图1.1 基本系统工程活动