可靠性·维修性·保障性技术丛书 ⑥

丛书主编 王自力

RMS

软件可靠性工程

Software Reliability Engineering

主编 陆民燕

Reliability
Maintainability
Supportability

国防工业出版社
National Defense Industry Press

可靠性·维修性·保障性技术丛书

软件可靠性工程

Software Reliability Engineering

主　　编　陆民燕

编写组成员　艾　骏　李秋英　吴玉美

（按姓氏笔画排序）　张　虹　曾福萍

国防工业出版社

·北京·

《可靠性·维修性·保障性技术丛书》编辑委员会

Preface 序 RMS

1995 年,国防科技及教育界著名专家杨为民教授组织编辑出版了国内第一套《可靠性·维修性·保障性丛书》,对推动武器装备质量观念的转变,提高武器装备的可靠性、维修性、保障性水平,发挥了重要的推动作用。

15 年后的今天,树立现代质量观,持续提高可靠性、维修性、保障性水平,已成为武器装备建设与国防科技发展中的共识,特别是《武器装备质量管理条例》的颁布实施,表明可靠性、维修性、保障性在现代质量观中具有战略性、全局性和基础性的地位和作用,高可靠、长寿命、好维修、易测试、能保障、保安全已成为武器装备研制、生产和使用中的普遍要求,可靠性、维修性、保障性工程活动已全面进入武器装备寿命周期各阶段,为提高武器装备的效能、降低寿命周期费用发挥了不可替代的作用。

在上述背景下,在武器装备建设与国防科技发展中,无论在技术上还是在管理上,都对可靠性、维修性、保障性提出了更高的要求。为适应这种新形势,我们组织有关专家重新编辑出版了这套《可靠性·维修性·保障性技术丛书》,共 12 册,以满足广大工程技术和管理人员的迫切需求。

本套丛书认真总结了 15 年来国内外武器装备可靠性、维修性、保障性最新实践经验,全面吸收了我国在预先研究和技术基础研究领域中取得的主要研究成果,从装备、系统、设备、元器件等多个产品层次和硬件、软件等不同产品类别,可靠性、维修性、测试性、保障性、安全性等多种质量特性,以及论证、研制、生产和使用与保障等寿命周期各阶段,全方位地论述了相关领域的基本概念、技术方法、实践经验及发展方向,具有系统性、实用性和前瞻性,从而有助于读者全面、系统地了解和掌握该项技术的全貌。本套丛书中阐述的可靠

性、维修性、保障性理论与技术，对武器装备和一般民用工业产品均具有普遍的适用性。

《可靠性·维修性·保障性技术丛书》是一套理论与工程实践并重的著作，它不仅可以为广大工程技术和管理人员提供有用的指导和参考，也可作为有关工程专业本科生、研究生的教学参考书。我们相信，这套丛书的出版，对我国武器装备可靠性、维修性、保障性工程的全面深入发展将起到重要的推动和促进作用。

丛书编辑委员会

2010 年 12 月

Preface 前言 RMS

本书的目的是帮助读者了解什么是软件可靠性工程，以及如何开展软件可靠性工程工作。在本书中，软件可靠性工程是指为了满足软件的可靠性要求而进行的一系列设计、分析、测试和管理等工作，它不仅包含了软件可靠性评估、软件可靠性预计、软件可靠性测试等与软件可靠性定量要求相关的工作，还包含了软件可靠性设计、分析等定性工作。特别指出这一点是因为在国外，对于软件可靠性工程更加强调的是定量的一面，如在最新的国际标准 IEEE 1633—2008 Recommended Practice on Software Reliability 中软件可靠性工程是指“应用统计技术处理在系统开发和运行期间所采集的数据，以便详细说明、预计、估计和评价基于软件的系统可靠性”。本书作者认为包含可靠性设计、分析、测试和管理等工作的软件可靠性工程观点更为系统和全面。

本书作者及其所在团队长期从事软件可靠性工程的科研工作，并结合工程实际开展应用研究工作。本书是作者及其所在团队十余年来研究成果的总结。此外，为了帮助读者了解并应用国际上比较成熟且得到广泛认可的软件可靠性工程技术，本书将最新的软件可靠性相关国际标准内容进行了引用和介绍。主要包括：IEEE Std 982.1™ – 2005 IEEE Standard Dictionary of Measures of the Software Aspects of Dependability、IEEE 1633 – 2008 Recommended Practice on Software Reliability、SAE(Society of Automotive Engineers)的相关标准 ——SAE JA1002 Software Reliability Program Standard(1998)，SAE JA1003 Software Reliability Program Guide(2004)。

本书主要是面向型号工程技术人员的，因此对阐述的技术和方法尽量给出示例、实施的注意事项等，以便于工程技术人员掌握、实施。本书的读者应具备软件开发基础、软件工程知识以及概率与数理统计基础。

全书内容共分 8 章。第 1 章：关于软件可靠性基本概念的概述。第 2 章：阐

述软件可靠性定量要求的确定方法，包括软件可靠性参数及其选取、指标的确定方法、软件可靠性定量要求确定过程等。基于 IEEE Std 982.1™—2005 推荐的若干面向过程评价和改进的软件可靠性度量。第 3 章：介绍多种软件可靠性分配方法以及测试之前的软件可靠性早期预计方法，后者是 IEEE 1633－2008 标准推荐的。第 4 章：叙述软件可靠性避错设计、容错设计方法，按软件需求分析阶段、软件设计阶段和软件实现阶段分别给出软件可靠性设计准则。第 5 章：阐明软件可靠性分析方法，主要包括软件 FMEA、软件 FTA。此外还介绍了软件 FMEA、软件 FTA 综合分析方法、事件树分析法和 Petri 网分析法。第 6 章：论述软件可靠性测试技术，包括软件可靠性增长测试、软件可靠性验证测试、软件可靠性摸底测试技术。给出了其中涉及的关键技术，主要包括软件操作剖面的构造、软件可靠性测试数据生成等技术。第 7 章：基于 IEEE 1633—2008，对典型的软件可靠性模型进行了介绍，包括呈指数分布的 NHPP 模型、非指数分布的 NHPP 模型及贝叶斯模型三类。给出了软件可靠性评估流程。第 8 章：关于软件可靠性管理方法的阐述。包括软件生存周期中的软件可靠性工程活动、软件可靠性评审、软件失效报告分析及纠正措施系统以及基于 SAE JA1002，SAE JA1003 的软件可靠性大纲管理框架。

为便于读者进一步了解软件可靠性早期预计的方法，附录 A 和附录 B 分别给出了 Keene 的开发过程预计模型（DPPM）和 SWEEP 软件缺陷早期预计，这也是 IEEE 1633—2008 在附录中推荐的早期预计模型。附录 C 给出了失效数据的趋势分析方法。

本书第 1 章、第 2 章、第 8 章由陆民燕教授编写，第 3 章～第 7 章由吴玉美、曾福萍、张虹、艾骏、李秋英撰写。全书由陆民燕教授主编。

本书由北京航空航天大学可靠性与系统工程学院的阮镰教授、装甲兵工程学院朱小冬教授主审。

为本书的编写提供材料和帮助的还有鲍晓红副教授、博士后付剑平、博士生李海峰、孟令中等。硕士生王学成为本书设计了方便实用的 WORD 模板。吴玉美协助完成了合稿等工作。

对于所有帮助和支持本书写作与出版的同志，我们表示深深的谢意！

由于水平和时间的限制，书中内容可能有错误和不当之处，请读者见谅并恳请提出宝贵意见。

编 者

2010 年 10 月

Contents 目录 RMS

第1章 绪 论

1.1 软件可靠性重要性

随着数字化设备大量应于装备,计算机软件在装备中的作用越来越大,其规模和重要性均呈急剧上升的趋势。

表 1-1 展示了美国第二、三、四代战斗机上,航电系统中由硬件和软件承担其功能的百分比,由表 1-1 可知:美国的战斗机每更新一代,其由软件实现的功能翻一番。航电系统软件规模也逐渐扩大,由最初的几千行发展到几百万行代码,据估计当前最先进的战斗机,如美国 F-35 的机载软件系统规模超过 600 万行源代码,F-22 的机载软件规模超过 200 万行。

表 1-1 软件在美国航电系统中的应用

第 X 代战斗机	型号	航电系统功能	
		硬件实现	软件实现
第二代	F-111	80%	20%
第三代	F-16	60%	40%
第四代	F-22	20%	80%

不幸的是装备软件的规模和复杂性剧增导致软件问题频出,缺陷数剧增。例如美国第四代战斗机 F-22 是美国空军第一个最复杂的软件密集型系统,航电系统软件规模超过 200 万行,美国空军一位高级官员总结 F-22 航空电子软件经验教训时说:“由于 F-22 战斗机航空电子软件很复杂,在飞机研制的初期,对航空电子软件的费用投入不足,造成了 F-22 飞行试验中出现了许多航空电子软件可靠性问题,拖延了飞行试验计划的进度。”

一项研究表明,专业的开发人员开发的软件平均每千行代码有 6 个缺陷,按照这一缺陷密度,一个 35 万行代码的软件系统缺陷数超过 2000 个。更为糟糕的是随着软件规模的增加缺陷密度呈几何级数增长,如图 1-1 所示。

表 1-2 还给出了一些软件的缺陷密度。

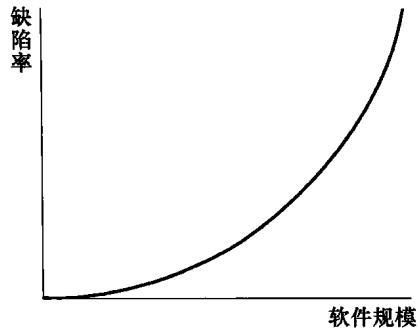

图 1-1 缺陷密度随规模变化示意图

表 1-2 一些软件应用程序的缺陷密度[1]

应用程序	数目	缺陷密度(100 LOC)
机载	8	1.28
战略	18	0.66
战术	6	1.00
过程控制	2	0.18
生产	9	1.30
开发	2	0.40

缺陷的增多导致缺陷定位和修复的成本大大增加。如微软定位并修改一个缺陷的平均时间相当于 12 个编程小时,那么调试一个 35 万行的软件需要 11.4 人年,费用超过 100 万美元。

由于软件的缺陷已经导致了大量的软件失效,其中一些产生了严重的后果。表 1-3 列出了曾被列为十大软件缺陷的事件。事实上,由于软件失效导致的严重事件远非仅此几项。这些惨痛的教训时刻警示我们,装备在投入实际使用之前必须考虑软件的可靠性。

表 1-3 十大软件缺陷[2]

编号	缺陷名称	缺陷描述	年份
10	马里纳 1 号空间探测器	无人驾驶的火箭偏离航向,以至于不得不被销毁。NASA 将此错误归因于 Fortran 语言的错误代码行(少写一个连字符)。成本损失 8000 万美元	1962
9	医疗放射伤害案	错误的软件和过程导致对癌症患者的致命的过量辐射	1985—1987
8	长途电话服务失效	呼叫处理电脑的交换机错误,导致长途电话网络瘫痪 9h,最终归因于一行错误代码	1990
7	爱国者导弹	追踪几枚飞毛腿导弹失效,其中一枚导致 28 名美军丧生。这个问题的原因是由于一个软件错误导致跟踪系统关闭 0.34s。事先已经发现了这个问题,但是没能及时修复从而导致美军的死亡	1991

（续）

编号	缺陷名称	缺陷描述	年份
6	奔腾芯片	对于特定的复杂方程，芯片给出错误答案。但是这个缺陷只影响了很小一部分英特尔用户。用于更换错误芯片的费用高达45000万美元	1994
5	金融机密泄露	有缺陷的税收软件，迫使制造商承诺及时支付造成的罚款和利息，最严重的漏洞允许某些用户访问制造商的主计算机上存储的报税表，并可以修改或删除	1995
4	丹佛机场关闭	行李处理系统的缺陷导致行李箱被损坏，自动行李车撞墙。16个月以后机场才恢复正常运营，为此多花了32亿美元，而且使用的是以手动为主的行李处理系统	1995
3	Java安全漏洞和浏览器崩溃	研究发现Java里的安全漏洞允许黑客从PC电脑里下载个人信息。互联网浏览器的争夺战所造成的加速开发进度导致了Java脚本溢出和自动重启等问题的出现	1996—1997
2	加州公共设施解除管制延迟	由于7天的新系统模拟发现了严重错误，必须推迟解除管制以调试运行电网的软件问题。为此，多花经费9000万美元，其中大部分将转嫁给纳税人，并可能迫使电力供应商负债累累或者停业	1998
1	千年虫:2000	在世纪交替时，此千年虫缺陷可能并没有导致很多问题，但是，它确实产生了大量的顾问公司和程序工具去处理这个问题	2000

美国的Koss博士曾指出："软件可靠性受到很大的忽视……，到1986年、1987年一些比较大的命令控制系统对软件可靠性不做任何评估。"他还指出，"在现代军用飞机上，软件已经超过100万行源代码，交付可靠的计算机硬件的