

JUNYONG RUANJIAN GONGCHENG XILIE BIAOZHUN SHISHI ZHINAN

军用软件工程 系列标准实施指南

总装备部电子信息基础部标准化研究中心 著

航空工业出版社

军用软件工程 系列标准实施指南

总装备部电子信息基础部标准化研究中心 著

航空工业出版社

内 容 提 要

GJB 5234《军用软件验证和确认》、GJB 5235《军用软件配置管理》等7项软件工程国家军用标准已于2004年9月20日正式颁布。该系列标准的制定,旨在通过软件工程标准化工作,有效提高我国军用软件质量。为了使军用软件的采购单位、承制单位等深入了解该系列标准的技术内容和实施方法,特编撰此书。

本书不仅从主要技术内容、理解要点及实施时应注意的问题等方面对新近发布的7项我国软件工程国家军用标准进行了详细说明,而且对军用软件工程标准体系、国内外软件工程标准化情况进行了全面介绍。本书是一本在军用软件工程标准化方面有指导性的辅导书,可供广大软件开发人员、软件测试人员以及软件管理人员使用。

图书在版编目(CIP)数据

军用软件工程系列标准实施指南/总装备部电子信息
基础部标准研究中心著. —北京:航空工业出版社,
2006. 2

ISBN 7-80183-706-1

I. 军… II. 总… III. 军用计算机—软件工程—
国家军用标准—中国—指南 IV. E919-62

中国版本图书馆CIP数据核字(2006)第003373号

军用软件工程系列标准实施指南

Junyong Ruanjian Gongcheng Xilie Biaozhun Shishi Zhinan

航空工业出版社出版发行

(北京市安定门外小关东里14号 100029)

发行部电话: 010-64919539 010-64978486

北京地质印刷厂印刷

全国各地新华书店经售

2006年2月第1版

2006年2月第1次印刷

开本: 787×1092 1/16

印张: 17

字数: 435千字

印数: 1—5000

定价: 58.00元

前 言

随着高新技术的发展和装备信息化程度的提高,软件在现代化武器装备中的比重不断增加,并已成为武器装备的重要组成部分,军用软件的质量直接影响甚至决定着武器装备的质量。为有效地保证和提高军用软件质量,总装备部电子信息基础部技术基础局于2003年组织开展了GJB 5234《军用软件验证和确认》、GJB 5235《军用软件配置管理》、GJB 5236《军用软件质量度量》、GJB 2434A《军用软件产品评价》、GJB 1268A《军用软件验收》等系列标准的制定、修订工作。

该系列标准已于2004年9月20日正式颁布,2005年1月1日起开始实施。为了帮助军用软件开发人员、管理人员及测评人员等在武器装备论证、研制、验收、运行等工作中正确理解和有效实施标准,特编撰了本教材。

本教材由总装备部技术基础管理中心和总装备部电子信息基础部标准化研究中心组织有关标准编制单位共同编写。本教材共分为7章和两个附录,其中第1章主要介绍了我国军用软件工程标准化情况以及该系列标准编制背景等,由潘华编写;第2章主要介绍软件生存周期的过程、软件生存周期模型及GJB 2786《武器系统软件开发》的有关要求,由王伟编写;第3章主要介绍GJB 5235《军用软件配置管理》,由石柱编写;第4章主要介绍GJB 5234《军用软件验证和确认》,由韩红强编写;第5章主要介绍GJB 5236《军用软件质量度量》,由冯惠编写;第6章主要介绍GJB 2434A《军用软件产品评价》,由王凌编写;第7章主要介绍GJB 1268A《军用软件验收》,由张为民编写;附录A介绍了国内外软件标准化情况,由潘华和刘远有编写;附录B介绍了软件生存周期模型特点、适用情况及其选择原则,由石柱编写。全书由傅兴男、丁树伟和仲斌负责统稿,程旭辉审阅。

在本教材的编写过程中得到了总装备部电子信息基础部技术基础局有关领导的大力支持和指导,得到了许聚常、尹平和张剑波等同志的大力帮助,在此表示衷心感谢。

由于时间仓促,加之水平和经验有限,教材中难免存在错误和不足之处,恳请读者批评指正。

编 者
2005年11月

目 录

第1章 引言	(1)
1.1 军用软件在武器装备中的地位	(1)
1.1.1 软件的特点	(1)
1.1.2 军用软件的作用和地位	(2)
1.2 军用软件工程标准化	(3)
1.2.1 软件工程的定义及内涵	(3)
1.2.2 软件工程标准化的作用	(5)
1.2.3 军用软件工程标准化状况	(6)
1.2.3.1 军用软件工程标准体系初步建立	(6)
1.2.3.2 对军用软件标准化重要性的认识不断深化	(7)
1.2.3.3 积极推进军用软件测评标准化工作	(7)
1.3 军用软件工程系列标准编制过程	(8)
第2章 军用软件生存周期过程	(11)
2.1 概述	(11)
2.1.1 定义和作用	(11)
2.1.1.1 软件生存周期	(11)
2.1.1.2 软件生存周期模型	(11)
2.1.1.3 软件生存周期模型的作用	(11)
2.1.2 软件生存周期过程	(12)
2.1.2.1 GB/T 8566—2001 规定的软件生存周期过程概貌	(12)
2.1.2.2 软件生存周期过程的活动	(12)
2.1.3 软件生存周期模型	(16)
2.2 GJB 2786—1996 的有关要求	(16)
2.2.1 术语	(16)
2.2.2 GJB 2786—1996 的基本特点	(17)
2.2.2.1 针对一个软件开发项目	(17)
2.2.2.2 把软件作为系统的一部分来对待	(17)
2.2.2.3 围绕软件开发工作将活动分类	(18)
2.2.3 GJB 2786—1996 的具体要求	(21)
2.2.3.1 各阶段的5类要求	(21)
2.2.3.2 各阶段5类要求的共同内容	(21)

2.2.3.3	各阶段 5 类要求的不同内容	(28)
2.2.3.4	向软件保障阶段转移	(35)
2.2.3.5	文档编制	(36)
2.2.4	GJB 2786—1996 的实施	(36)
2.2.4.1	参照相应的标准	(37)
2.2.4.2	软件的获取和维护保障	(37)
第 3 章	GJB 5235 《军用软件配置管理》	(41)
3.1	概述	(41)
3.1.1	编制原则	(41)
3.1.2	本标准与其他标准的关系	(41)
3.1.2.1	与 ISO 12207 和 GB 8566 的关系	(41)
3.1.2.2	与 GJB 5000 的关系	(42)
3.1.2.3	与 GJB 2786—1996 的关系	(46)
3.2	基本概念	(48)
3.2.1	软件配置项	(48)
3.2.2	软件配置管理	(49)
3.2.3	基线	(50)
3.2.4	软件库	(50)
3.3	配置管理的一般要求	(51)
3.4	配置管理过程	(52)
3.5	配置标识	(55)
3.6	配置控制	(57)
3.6.1	访问控制	(57)
3.6.2	版本控制	(58)
3.6.3	检入和检出控制	(58)
3.6.4	更改控制	(59)
3.7	配置状态记实	(62)
3.8	配置评价	(62)
3.9	软件发行管理和交付	(62)
3.10	实施 GJB 5235 应注意的问题	(67)
第 4 章	GJB 5234 《军用软件验证和确认》	(68)
4.1	概述	(68)
4.1.1	GJB 5234 的编制背景	(68)
4.1.2	GJB 5234 的编制原则	(68)
4.1.3	国内外相关标准	(68)
4.2	本标准与其他国家标准和军用标准的关系	(69)
4.3	基本概念	(69)
4.4	软件完整性级别	(69)

4.4.1	关键软件与关键性分析	(69)
4.4.2	指定软件完整性级别	(70)
4.5	过程、活动与任务	(71)
4.5.1	V&V 过程	(71)
4.5.1.1	管理过程	(71)
4.5.1.2	获取过程	(72)
4.5.1.3	供应过程	(72)
4.5.1.4	开发过程	(72)
4.5.1.5	运作过程	(74)
4.5.1.6	维护过程	(75)
4.5.2	常用的 V&V 活动	(75)
4.5.2.1	可追踪性分析	(76)
4.5.2.2	评价	(76)
4.5.2.3	接口分析	(77)
4.5.2.4	测试	(78)
4.5.3	执行 V&V 任务	(81)
4.6	软件验证和确认计划 (SVVP) 的制定	(82)
4.6.1	目的	(83)
4.6.2	参考文档	(83)
4.6.3	定义	(83)
4.6.4	验证和确认概述	(83)
4.6.4.1	组织	(84)
4.6.4.2	主进度	(84)
4.6.4.3	资源摘要	(84)
4.6.4.4	职责	(85)
4.6.4.5	工具、技术和方法	(85)
4.6.5	生存周期验证和确认	(86)
4.6.5.1	V&V 管理	(86)
4.6.5.2	概念 V&V 活动	(88)
4.6.5.3	需求 V&V 活动	(89)
4.6.5.4	设计 V&V 活动	(91)
4.6.5.5	实现 V&V 活动	(93)
4.6.5.6	测试 V&V 活动	(96)
4.6.5.7	安装和检验 V&V 活动	(97)
4.6.5.8	运行和维护 V&V	(99)
4.6.6	V&V 报告	(101)
4.6.6.1	要求的报告	(101)
4.6.6.2	可选的报告	(102)
4.6.7	验证和确认管理规程	(102)
4.6.7.1	异常报告的解决方案	(102)

4.6.7.2	任务重复策略	(104)
4.6.7.3	偏离策略	(104)
4.6.7.4	控制规程	(105)
4.6.7.5	标准、惯例和约定	(106)
4.7	支持软件验证和确认的技术与方法	(107)
4.7.1	选择软件 V&V 技术的策略	(107)
4.7.2	审查	(108)
4.7.3	V&V 度量	(108)
4.8	独立验证和确认 (IV&V)	(109)
第 5 章	GJB 5236 《军用软件质量度量》	(111)
5.1	概述	(111)
5.1.1	质量度量标准编制背景及编制原则	(111)
5.1.2	GJB 5236 的编制说明	(111)
5.1.3	软件产品质量标准的演变与发展	(112)
5.1.4	GJB 5236 与 GJB 2434A 的关系	(112)
5.1.5	GJB 5236 和 GJB 2434A 的实施需关注的问题	(113)
5.1.5.1	基本要求	(113)
5.1.5.2	符合性	(113)
5.1.5.3	适用性	(114)
5.1.5.4	软件质量度量标准与软件生存周期过程标准的关系	(114)
5.2	软件质量度量基本概念	(115)
5.2.1	软件质量	(115)
5.2.2	软件质量要求	(116)
5.2.3	软件质量度量	(118)
5.2.3.1	软件度量的目的	(118)
5.2.3.2	软件度量的对象	(119)
5.2.3.3	客观度量和主观度量	(119)
5.2.3.4	直接度量和间接度量	(119)
5.2.4	软件质量的标度	(120)
5.2.4.1	标称标度	(121)
5.2.4.2	顺序标度	(121)
5.2.4.3	间隔标度	(122)
5.2.4.4	比率标度	(122)
5.2.4.5	绝对标度	(123)
5.2.5	提高软件质量的途径	(124)
5.3	软件质量模型	(125)
5.3.1	内部质量和外部质量模型	(125)
5.3.2	内部质量和外部质量的分层定义	(125)
5.3.3	使用质量的质量模型	(129)

5.4 质量模型的使用	(130)
5.5 度量表的阅读和使用	(132)
5.6 度量元度量示例说明	(133)
5.7 软件度量的一些认识	(135)
5.7.1 注意事项	(135)
5.7.2 期望值要现实	(135)
5.7.3 软件度量原则	(135)
5.7.4 软件度量的误区	(136)
5.8 软件测量与评价实例(某网络通信服务软件)	(136)
5.9 小结	(139)
5.10 国际标准未来发展动向——SQuaRE 系列标准概述	(140)
5.10.1 SQuaRE 的体系结构	(140)
5.10.2 ISO/IEC 9126、ISO/IEC 14598 与 SQuaRE 系列标准的对应关系	(141)
5.10.3 SQuaRE 通用质量模型	(142)

第6章 GJB 2434A《军用软件产品评价》

(144)

6.1 概述	(144)
6.1.1 编制目的和意义	(144)
6.1.2 编制背景	(145)
6.1.3 编制原则	(145)
6.1.4 与其他标准的关系	(146)
6.2 基本概念	(146)
6.2.1 软件质量和质量特性	(146)
6.2.2 软件质量管理和软件产品评价	(147)
6.2.3 评价级别	(148)
6.3 软件产品的一般评价过程	(149)
6.3.1 确立评价需求	(150)
6.3.1.1 确立评价目的	(150)
6.3.1.2 标识产品类型	(151)
6.3.1.3 规定质量模型	(151)
6.3.2 规定评价	(153)
6.3.2.1 选择度量	(153)
6.3.2.2 确立质量评定等级	(154)
6.3.2.3 确立评估准则	(154)
6.3.3 设计评价	(154)
6.3.4 执行评价	(154)
6.4 开发者用的评价过程	(156)
6.4.1 适用范围	(156)
6.4.2 开发者用的评价概念	(157)
6.4.3 评价过程的实施	(158)

6.4.3.1	确立评价需求	(158)
6.4.3.2	规定评价	(158)
6.4.3.3	设计评价	(159)
6.4.3.4	执行评价	(159)
6.4.3.5	质量评价的评审和对组织的反馈	(160)
6.5	需方用的评价过程	(160)
6.5.1	适用范围	(160)
6.5.2	需方用的评价的概念	(160)
6.5.2.1	确立评价需求时应注意的问题	(161)
6.5.2.2	在设计评价时应注意的问题	(161)
6.5.2.3	评价的剪裁	(162)
6.5.3	评价过程的实施	(162)
6.5.3.1	获取现货软件产品期间的评价	(163)
6.5.3.2	定制软件的获取和修改现有的软件期间的评价	(169)
6.6	评价者用的评价过程	(171)
6.6.1	适用范围	(171)
6.6.2	评价者用的评价的概念	(172)
6.6.2.1	评价的原则	(172)
6.6.2.2	主要关注的内容	(172)
6.6.3	评价者用的评价过程的实施	(173)
6.6.3.1	职责	(173)
6.6.3.2	确立评价需求	(174)
6.6.3.3	规定评价	(174)
6.6.3.4	设计评价	(174)
6.6.3.5	执行评价	(175)
6.6.3.6	作出评价结论	(176)
6.7	评价的策划和管理	(181)
6.7.1	一般考虑和支持机构的主要任务	(181)
6.7.2	评价的管理支持	(182)
6.7.2.1	评价策划的组织级的管理支持	(182)
6.7.2.2	评价策划的项目级的管理支持	(182)
6.8	评价模块的文档编制	(183)
6.8.1	评价模块的概念	(183)
6.8.2	评价模块的文档结构	(183)
6.8.3	评价模块的开发	(184)
6.8.4	评价模块的示例	(184)
第7章	GJB 1268A《军用软件验收要求》	(199)
7.1	概述	(199)
7.1.1	编制背景和目的	(199)

7.1.2	编写原则	(199)
7.1.3	国内外标准状况	(200)
7.1.4	与其他相关国军标、国标、国外先进标准的关系	(200)
7.1.5	GJB 1268A 与军用软件定型管理办法的关系	(200)
7.1.6	对原标准的分析	(201)
7.1.7	本修订标准与前版本的主要差异	(201)
7.2	基本概念	(202)
7.2.1	军用软件	(202)
7.2.2	软件验收	(202)
7.2.3	验收测试	(202)
7.2.4	验收准则	(202)
7.2.5	软件完整性级别	(202)
7.2.6	现货软件	(203)
7.2.7	合同	(203)
7.2.8	委托开发软件	(203)
7.2.9	验收审查	(203)
7.2.10	验收评审	(203)
7.3	主要内容及技术要点	(204)
7.3.1	范围	(204)
7.3.2	概要	(204)
7.3.3	一般要求	(205)
7.3.3.1	软件验收前提	(205)
7.3.3.2	软件验收各方职责	(205)
7.3.3.3	软件验收依据	(206)
7.3.3.4	软件验收程序	(206)
7.3.4	详细要求	(206)
7.3.4.1	软件验收申请	(206)
7.3.4.2	被验收软件及相关文档的提交	(208)
7.3.4.3	软件验收计划	(209)
7.3.4.4	软件验收组织	(212)
7.3.4.5	软件验收测试和验收审查	(213)
7.3.4.6	软件验收评审	(218)
7.3.4.7	软件验收结论	(220)
7.3.5	对 GJB 1268A 的裁剪	(220)
7.3.5.1	一般考虑	(220)
7.3.5.2	高完整性级别软件	(220)
7.3.6	文档格式	(221)
7.4	标准贯彻实施应当注意的问题	(221)
7.4.1	GJB 1268A 对验收依据 (合同或双方约定的验收依据的文档) 提出了 要求	(221)

7.4.2	军用软件的分类	(221)
7.4.3	应用 GJB 1268A 的时间切入点问题	(222)
7.4.4	软件完整性级别	(223)
7.4.5	如果验收依据中没有规定软件完整性级别如何处理	(223)
7.4.6	验收不通过的处理	(223)
7.4.7	验收方对验收结论不满的处理	(223)
7.4.8	验收通过之后的软件移交	(223)
附录 A	国内外软件工程标准状况	(225)
A1	国外软件工程标准状况	(225)
A1.1	ISO 软件工程标准现状	(225)
A1.2	IEEE 软件工程标准现状	(230)
A1.3	美国军用软件工程标准化状况	(232)
A1.4	欧洲软件工程标准化状况	(240)
A2	国内软件工程标准状况	(242)
A2.1	国家软件工程标准状况	(242)
A2.2	国防科技工业各行业标准	(246)
附录 B	软件生存周期模型及其选择原则	(250)
B1	基本概念	(250)
B2	瀑布模型	(250)
B3	增量模型	(252)
B4	进化模型	(254)
B5	基于软件包的生存周期模型	(255)
B6	遗留系统维护生存周期模型	(257)
B7	软件生存周期模型选择原则	(258)

第1章 引言

20 世纪 40 年代，美国研制出了世界上第一台计算机，并编制了用于研制原子弹和计算导弹弹道的军用软件。从那时到今天，军用软件的发展不过近 60 年的历程，却已在军事领域的各个方面得到深入广泛的应用，成为了各种高新武器装备不可缺少的重要组成部分，在某些领域，软件本身已成为一类重要的装备。军用软件的发展及应用水平，成为军队信息化程度的主要标志，从某种角度上甚至可以说代表着一个国家的军事实力。因此，世界各主要国家和地区都把军用软件作为推进军队信息化的重要途径。

然而，由于军用软件本身所固有的高复杂性、研制上的高风险性及管理上的高难度，使军用软件的发展一直在曲折中前进，军用软件质量和生产率无法满足军事要求、预算严重超支、项目延期或中途下马等事件屡见不鲜。军用软件工程标准化由于其在有效提高军用软件质量方面发挥了重要作用而迅速发展起来。

1.1 军用软件在武器装备中的地位

1.1.1 软件的特点

软件是指与操作一个计算机有关的计算机程序、进程以及可能相关的记录和数据。软件的工作是告诉计算机做什么和如何做。没有软件，计算机就成为只有机箱和屏幕的盒子，甚至许多简单的家用电器如微波炉、洗衣机等都会成为废物。

软件没有“质量”，看不见，摸不着，无法称重量，正因为如此，软件常常被误解、忽视或者与硬件混为一谈。因为它没有物理特性，人们难以理解在微妙电磁场世界中以字和字节形式存在的看不见的事物。

因为软件是无形的，所以它不仅难以理解和描述，而且难以制作，不能用物理方法进行制造。软件具有与硬件明显不同的特点。

a) 软件是被开发或设计的，而不是传统意义上被制造的。虽然在软件开发和硬件制造之间有一些相似之处，但两类活动在本质上是不同的。对于这两者都可以通过良好的设计得到高质量，但硬件在制造过程中可能引入质量问题，而对于软件几乎不存在或很容易改正。两者都是要建造一个“产品”，但方法不同，硬件几乎都可以自动生产，但是生产软件的方法却是极其低效，基本上只能手工劳动生产。软件作为人的智力产品，易受人为错误的影响，其开发过程不透明，易于修改，质量难以控制和考核；而硬件的生产过程可以通过严格的生产规程来控制，外部质量可通过仪器设备来检测。软件成本集中于开发上，因此软件项目也不能像制造项目那样管理。

b) 软件不会“磨损”。硬件故障率与时间的函数常称为“浴缸曲线”，表明硬件初期由

于设计或制造的缺陷有较高的故障；这些缺陷修正之后，故障率相对稳定一段时间。随着时间的流逝，各种环境的侵蚀，故障率又提升了。简言之，硬件开始磨损了。软件并不受这些环境因素的影响，在初期由于未发现的错误会引起程序具有较高的故障率，当错误改正后，理想中故障率应趋于平稳，但由于这些修改有可能会引入新的错误，故障率曲线呈现锯齿形，最小故障率水平也开始提高。实际上软件的退化是由于修改。

c) 复杂性是软件的一个固有特性。在其他学科中常常用简化技术，而软件则不是这样，软件的本质是通过合成复杂性来获得解决复杂问题的方法，其定义的解决方法远比要解决的现实世界中的复杂问题还要复杂得多。同时，软件作为信息产品，相互之间的接口状态复杂多变，集成困难；而硬件的接口相对简单，比较容易规范，集成组装相对容易等。另外，当一个硬件构件磨损时，可以用另外一个备用零件替换它，而软件故障常常表明设计中出现错误，因此软件维护要比硬件复杂得多。

1.1.2 军用软件的作用和地位

军用软件通常是指用于军事目的的一类软件，一般可分为两大类。一类是武器系统软件，另一类是非武器系统软件（称为自动化信息系统软件）。武器系统软件包括为武器系统专门设计或专用的并成为整个系统不可缺少的一部分嵌入式软件；指挥、控制和通信软件；对武器系统及其完成任务起保障作用的其他武器系统软件，如任务规划软件、战斗管理软件、后勤保障软件、演习分析软件、训练软件、飞行计划软件、应用测试软件、程序管理软件、模拟器软件等。非武器系统软件（称为自动化信息系统软件）主要指执行与武器系统无关的系统使用和保障功能的软件，例如，科学计算、人员管理、资源控制、地图管理、设备维修、仿真、人工智能软件等。

随着军队信息化程度的不断提高，军用软件已经渗透到军事应用的各个方面，已成为武器装备体系中不可或缺的组成部分。当现代武器系统被称为“智能”武器时，则是因为软件为其提供了大脑。软件通过专用硬件的运行，可以完成许多的功能，如作战飞机的每一次使用基本上都依赖于软件，包括战略和战术行动；监视、探测、评估和预警等。在不少情况下，光有硬件是不行的，软件的功能甚至要超出其他部件，如软件控制所有飞机的垂直稳定度，软件使飞机的隐身技术成为可能。软件的重要性已在最近的几次高技术局部战争中得到证明，主要体现在如下几个方面。

a) 软件是高新武器装备的灵魂

在高新武器装备中由软件实现的功能越来越多，有些装备甚至大大超过了硬件；而且软件不但能执行以前由硬件执行的许多功能，而且还能执行光靠硬件几乎无法执行的功能，例如，为减少雷达截面积，B-2 轰炸机没有垂直控制面，飞机的垂直稳定度全靠软件来控制，从而满足了 B-2 轰炸机隐身的要求。软件的应用不仅极大地提高了武器装备系统原有的许多性能，而且已成为整个军事系统的控制中枢和威力倍增器，成为高新武器装备的灵魂。

b) 软件是构筑信息化装备体系的关键

现代战争是体系和体系的对抗，单一武器、单一系统的决胜作用已经逐渐弱化，而由大量嵌入芯片和软件的各种武器装备形成的信息化装备体系已成为战争制胜的基础。构筑信息化装备体系绝不是各种装备的简单堆积，也不只是各种武器装备和系统之间的物理连通，而是通过软件的控制，使各种作战信息按照作战要求有序流动，满足体系内各部分之间互连互

通互操作的要求,实现不同武器系统的功能互补、协同行动和互相支援。因此说软件已成为构筑信息化装备体系的关键。

c) 软件可有效提升武器装备的整体作战效能

由于软件能极大提高武器系统的信息获取、传输、处理、存储、管理、分发及其数字化、智能化、网络化水平,利用软件技术对已有武器装备进行改造已成为部分武器装备升级换代所依靠的主要模式之一。利用软件技术改造不仅能大大提高武器装备作战效能,而且还具有成本小、周期短和效果好的优点。例如,伊拉克战争前夕,美国海军 F-14 战斗机主要通过软件升级的方式具备了投放精确制导武器的能力,使改造周期从以往的几个月甚至几年,缩短至 3 个星期;另外,美军的全球指挥控制系统近几次的改进也主要是通过软件升级而实现的。

d) 软件是信息战中攻防对抗装备发展的焦点

信息战已成为现代战争的主要模式。一方面,作为一类特殊软件——计算机病毒成为信息战进攻的重要手段。计算机病毒的作用主要是通过破坏敌方信息系统和计算机网络中的软件,达到使其瘫痪的目的。海湾战争中,美军利用开发的计算机病毒“预埋”技术,在芯片中置入计算机病毒,给伊方造成了重大的损失。目前,“台军”已把计算机病毒作为反大陆的重要手段,并在实战中进行演练。另一方面,信息安全软件、网络防护软件成为信息战防御的关键工具。为对抗计算机病毒和其他针对信息系统及网络中的软件攻击,保证其正常运行,各国都在开发以软件为主要形式的对抗措施。可以预见,随着信息系统及网络的广泛应用,以争夺信息优势为目标的信息对抗日趋激烈,军用软件将成为敌我双方信息攻防对抗装备的焦点。

1.2 军用软件工程标准化

1.2.1 软件工程的定义及内涵

软件本身的复杂性、抽象性和易变性,使软件难以研制,质量难以保证,而军用软件的特殊应用又使其开发和质量保证难度增大,主要体现在以下几个方面。

a) 军用软件要面临复杂、不确定和恶劣的作战环境,要求军用软件必须具有抗毁和容错能力,因此,军用软件应具有高可靠性、高安全性和高生存性;

b) 在信息对抗环境下要求军用软件具有较高的安全防护能力,因此,军用软件应具有高保密性;

c) 作战使命任务对军用软件的信息传输处理速度、对外部事件的快速响应提出了更高的要求,因此,军用软件应具有高的实时性要求;

d) 军用软件不少是嵌入式的,受到严格的硬件和软件条件的约束,被硬件及软件体系结构、操作系统特性、应用需求和编程语言的变化所制约;

e) 同时,军用软件的开发还要纳入武器装备研制过程,这决定了军用软件开发方法要与硬件研制中采用的自下而上的方法相一致;

f) 军用软件规模巨大,如一架现代战斗机所包含的软件已经超过 2500 万行源代码,一艘现代化战舰或潜水艇所包含的软件有 5000 万行源代码,软件规模越大就越复杂,这不仅

带来了技术问题，而且为软件的管理带来了很大的困难；

g) 现代化战争是一体化的联合作战，需要进行数据交换、信息共享、应用协同，这些均对军用软件提出了高互操作性要求。

上述军用软件的特点对军用软件研制管理的方法和技术都有特殊要求，同时还要求开发人员应具备相关应用领域知识，另外，对开发设施和工具、安全保密方面都有较高的要求。

因此，早在 20 世纪 60 年代末，美国国防部门和软件界有关人士首先提出了“软件危机”的警告。软件危机主要体现在：

a) 难以正确估计软件开发的成本和进度，软件研制成为武器装备研制进度延误、研制经费超支和物资浪费的主要因素；

b) 软件质量不高、不可靠，用户经常不满意“已完成的”软件系统；

c) 没有适当的文档来记录软件开发过程中的信息及变化，软件常常不可维护；

d) 软件生产率越来越低等。

长期以来软件危机一直存在着，几乎每一个重大项目的失败，问题往往都可以归结到风险最高的部件——软件，软件问题已成为引起武器系统交付延迟、系统误动作的主要原因。从表 1-1 中可以看出，软件的研制失败率，尤其是大型软件项目的研制失败率是很高的。美国国防部软件研究计划的负责人回顾 82 个大型军事采办计划时，他就发现大多数软件研制项目比计划进度推迟 20 个月完成，是非密集软件项目计划的 3 倍，由于这些项目的延期花去国防部当年研究和采购预算费用的十分之一，该负责人还说：“国防部为未能解决软件问题受到了巨大的惩罚，惩罚不仅是软件的时间推迟，而且使作战能力大打折扣。”

表 1-1 1994 年美国《科学》杂志软件业调查结果（大型软件研制计划）

%

调查项目	调查结果
被取消的计划	33
进度拖延	50
作战失败	75

通过研究，人们普遍认为软件危机的根本原因是软件开发和生产过程采用“手工作坊”模式，软件开发和生产过程不透明，缺乏有效的管理。显然解决软件危机仅靠技术措施是不够的，还必须采用先进的管理措施。因此，解决软件危机的一个主要出路是：① 改变过去那种手工作坊式的开发方式，采用与机械工程类似的工程方法，按照工程化的概念、原理、技术和方法来组织软件开发工作；② 推广使用在实践中总结出来的、成功的软件开发技术和方法，研究探索更为有效的技术和方法，以提供关于软件开发的一般原则、工作框架、开发策略和若干实用技术；③ 开发并使用行之有效的软件开发工具和环境，以提高软件开发人员的工作效率，减少出现人为差错的可能性。“软件工程”一词在 1968 年北约举行了一次软件学术会议上首次正式提出，并随着软件生产的发展和软件开发技术的不断完善，而逐渐形成了一个新的工程分支。

根据 GB/T 11457—1995《软件工程术语》中“软件工程”定义为“软件开发、运行、维护和引退的系统方法”，目的就是为软件全生存周期活动提供工程化的手段，从而提高软

件的质量、降低成本和缩短开发周期等。

软件工程主要包括四个要素：方法与技术、工具与环境、管理与控制、标准与规范。其中：

a) 软件工程方法与技术提供关于软件开发的一般原则、工作框架、开发策略和若干实用技术。其中包括软件生存周期模型、自顶向下方法、结构化开发方法、面向对象的方法、需求工程和测试技术等。

b) 软件工程工具与环境为软件工程方法提供自动或半自动的软件支撑环境。它可用于提高软件的生产率，保证软件质量，便于软件测试和集成，提高软件开发过程的可见性和可控性。

c) 软件工程管理与控制是将软件工程的方法和工具综合起来，以达到合理、及时地进行计算机软件开发的目的。软件工程管理主要包括项目、配置、文档、质量、经费、人员与进度等方面的内容。

d) 软件工程标准与规范是为软件开发和管理的过程以及软件产品规定的共同准则，它通常包括术语和符号标准、产品标准、方法和技术标准以及管理标准等。

1.2.2 软件工程标准化的作用

软件工程标准化的进程是同软件工程化的发展水平相适应的。在 20 世纪 60 年代，随着计算机的进步，涌现了大量的程序设计语言，即使是同一种语言由于在不同型号上实现时作了不同程度的修改，也会为软件的移植造成很大的障碍。为了解决程序设计语言的不统一问题，人们制定了标准的程序设计语言，这便是软件工程标准化的开始。在软件工程化推动下，软件工程标准化的范围也从程序设计语言扩展到软件的开发、软件的维护直至软件整个生存周期的技术和管理，软件开发标准、软件文档标准、软件质量保证标准、软件验证和确认标准、软件生存周期过程标准等应运而生。

简言之，软件工程标准化就是通过制定、贯彻并监督实施标准，规范软件开发、运行、维护和引退全过程工作和产品，以提高软件产品质量。其作用具体体现在：

a) 通过开展软件工程标准化，可以为软件工程活动规定通用的框架和基本要求，有助于保证软件工程活动的完整性、有效性，提高管理的透明度、可控性和有序性；

b) 通过开展软件工程标准化，可以为软件开发的各单位或人员规定共同的行为准则，有助于协调和统一软硬件研制活动；

c) 通过开展软件工程标准化，把选定的软件工程方法变为具体规定，有助于克服由多种方法学并用所带来的困难，保证软件开发方法与硬件研制的方法相协调；

d) 通过开展软件工程标准化，可以统一软件产品可能具有的共同或相近的属性并用以指导设计和开发，有利于提高软件的重用率、互操作性、保障性和综合集成能力；

e) 通过开展软件工程标准化，可以规定检验软件开发工作成果的共同依据，有助于软件的评审、测试和验收等。

总之，通过软件工程标准化工作，可以大大提高软件的可靠性、安全性、可维护性、生产率、可移植性，促进软件的重用，从而有效保证产品质量、降低全生存周期费用、缩短开发周期及部署时间、提高综合保障能力。