



中华人民共和国国家标准化指导性技术文件

GB/Z 31102—2014

软件工程 软件工程知识体系指南

Software engineering—Guide to the software engineering body of knowledge
(SWEBOK)

(ISO/IEC TR 19759:2005, MOD)

2014-09-03 发布

2015-02-01 实施

中华人民共和国国家质量监督检验检疫总局
中国国家标准化管理委员会 发布

目 次

前言	V
引言	VI
1 范围	1
2 软件需求	1
2.1 综述	1
2.2 软件需求基础	2
2.3 需求过程	3
2.4 需求引出	4
2.5 需求分析	6
2.6 需求规格说明	8
2.7 需求确认	9
2.8 实践考虑	10
2.9 主题与参考文献对照表	11
2.10 对软件需求推荐的参考文献	13
2.11 进一步阅读的参考文献	13
2.12 相关标准	18
3 软件设计	19
3.1 综述	19
3.2 软件设计基础	20
3.3 软件设计的关键问题	21
3.4 软件结构与体系结构	22
3.5 软件设计质量分析与评价	23
3.6 软件设计记法	24
3.7 软件设计策略与方法	25
3.8 主题与参考文献对照表	26
3.9 对软件设计推荐的参考文献	31
3.10 进一步阅读的参考文献	32
3.11 相关标准	33
4 软件构造	33
4.1 综述	33
4.2 软件构造基础	34
4.3 构造管理	35
4.4 实践考虑	36
4.5 主题与参考文献对照表	39
4.6 对软件构造推荐的参考文献	40
4.7 进一步阅读的参考文献	40
4.8 相关标准	40
5 软件测试	41

- 5.1 综述 41
- 5.2 软件测试基础 42
- 5.3 测试级别 44
- 5.4 测试技术 47
- 5.5 测试相关测度 50
- 5.6 测试过程 52
- 5.7 主题与参考文献对照表 54
- 5.8 对软件测试推荐的参考文献 58
- 5.9 进一步阅读的文献 58
- 5.10 相关标准 60
- 6 软件维护 60
 - 6.1 综述 60
 - 6.2 软件维护基础 61
 - 6.3 软件维护的关键问题 63
 - 6.4 维护过程 67
 - 6.5 维护技术 70
 - 6.6 主题与参考文献对照表 70
 - 6.7 对软件维护推荐的参考文献 75
 - 6.8 进一步阅读的参考文献 76
 - 6.9 相关标准 78
- 7 软件配置管理 78
 - 7.1 综述 78
 - 7.2 SCM 过程管理 80
 - 7.3 软件配置标识 84
 - 7.4 软件配置控制 86
 - 7.5 软件配置状态记述 88
 - 7.6 软件配置审核 88
 - 7.7 软件发布管理与交付 89
 - 7.8 主题与参考文献对照表 90
 - 7.9 对软件配置管理推荐的参考文献 92
 - 7.10 进一步阅读的参考文献 92
 - 7.11 相关标准 93
- 8 软件工程管理 93
 - 8.1 综述 93
 - 8.2 初启与范围界定 95
 - 8.3 软件项目规划 96
 - 8.4 软件项目实施 98
 - 8.5 评审与评价 99
 - 8.6 闭合 99
 - 8.7 软件工程测量 100
 - 8.8 主题与参考文献对照表 101
 - 8.9 对软件工程管理推荐的参考文献 103
 - 8.10 进一步阅读的参考文献 103

8.11	相关标准	107
9	软件工程过程	107
9.1	综述	107
9.2	过程实现与变更	108
9.3	过程界定	110
9.4	过程评估	111
9.5	过程与产品测量	112
9.6	主题与参考文献对照表	115
9.7	对软件工程过程推荐参考的文献	119
9.8	进一步阅读的参考文献	120
9.9	相关标准	123
10	软件工程工具与方法	124
10.1	综述	124
10.2	软件工程工具	126
10.3	软件工程方法	128
10.4	主题与参考文献对照表	129
10.5	对软件工程工具与方法推荐的参考文献	130
10.6	进一步阅读的参考文献	131
10.7	相关标准	132
11	软件质量	132
11.1	综述	132
11.2	软件质量基础	133
11.3	软件质量管理过程	135
11.4	实践考虑	138
11.5	主题与参考文献对照表	142
11.6	对软件质量推荐的参考文献	143
11.7	进一步阅读的参考文献	145
11.8	相关标准	146
12	软件工程相关学科	147
12.1	导引	147
12.2	计算机工程	148
12.3	计算机科学	149
12.4	管理	149
12.5	数学	149
12.6	项目管理	150
12.7	质量管理	150
12.8	软件工效学	150
12.9	系统工程	152
附录 A (资料性附录)	IEEE 与 ISO 软件工程标准对 SWEBOK 知识域的分配	153
附录 B (资料性附录)	根据 Bloom 的分类法进行的主题分类	161
附录 C (资料性附录)	本指导性技术文件与 ISO/IEC TR 19759:2005 相比的结构变化情况	172

前 言

本指导性技术文件按照 GB/T 1.1—2009 给出的规则起草。

本指导性技术文件使用重新起草法修改采用 ISO/IEC TR 19759:2005《软件工程 软件工程知识体系指南》。

本指导性技术文件与 ISO/IEC TR 19759:2005 的结构性差异参见附录 C。

本指导性技术文件由全国信息技术标准化技术委员会(SAC/TC 28)提出并归口。

本指导性技术文件起草单位:广西达译商务服务有限公司、中国电子技术标准化研究院、中国科学院自动化研究所、上海宝信软件股份有限公司。

本指导性技术文件主要起草人:温家凯、张旻旻、王力、邓姿娴、刘连芳、施贵基、丁锦春、陆迪茜、丛力群、滕逸龙、韦寒梅、宗成庆、周玉。

引 言

本指导性技术文件的读者范围广泛。主要对象是,需要在软件工程的如下方面取得一致观点的各种组织:确定教育和培训需求、工作分类、制定绩效评估方针、规定软件开发任务等。也面向执业(或管理)软件工程师以及负责制定公共许可政策及职业指南的公职人员。另外,如下人员也将从本指导性技术文件中受益:为大学课程界定认证准则、合格评定方针以及职业实践指南的协会和教师,学习软件工程的学生,制定课程表及课程内容的教师和培训人员。

本指导性技术文件将软件工程学科内容划分为 10 个知识域(KA)。第 2 章~第 11 章分别对应这 10 个知识域。建立学科界线时,重要的是确定软件工程与什么学科共享边界以及有公共交集。为此,本指导性技术文件还认定了八个相关学科(见第 12 章“软件工程相关学科”)。

本指导性技术文件采用分解结构,将知识域分解成若干子域,子域又分为若干主题。这为读者找到感兴趣的主体提供了适当的途径。

图 1 为软件工程知识体系指南的知识域分解图。

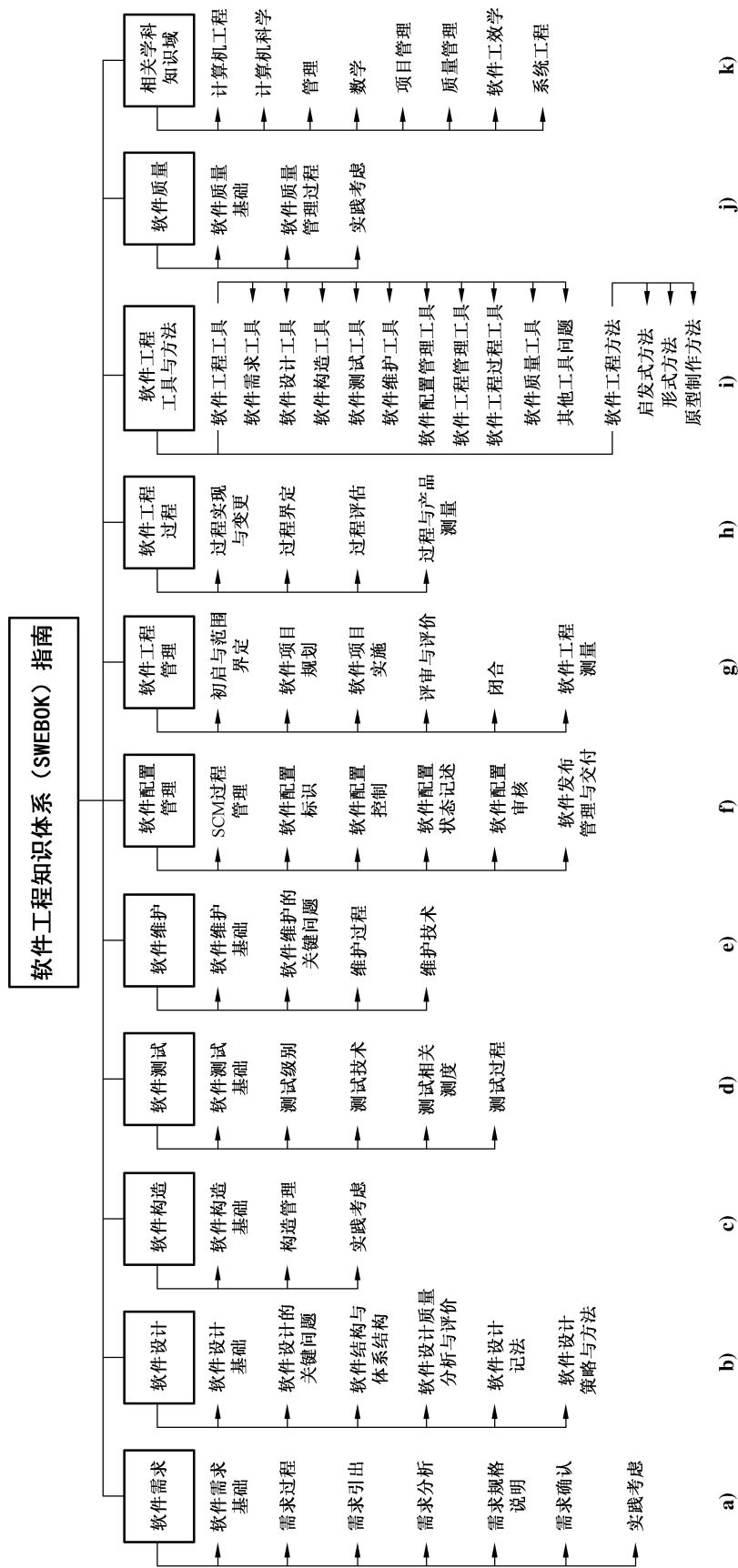


图 1 知识域分解图

软件需求、软件设计、软件构造、软件测试、软件维护这五个知识域按传统的瀑布生存周期顺序进行描述,但是,这并不表示本指导性技术文件采纳或鼓励瀑布模型或任何其他模型。软件配置管理、软件工程管理、软件工程过程、软件工程工具与方法、软件质量这五个知识域按字母顺序进行描述。

本指导性技术文件为各知识域附上了各种参考文献,包括图书章节、指定论文或其他认可的权威信息来源。对每一知识域的描述还包括参考文献与所列主题联系对照表。

知识域描述结构

知识域描述的结构如下:

引言中提出了知识域的简明定义,并概述了知识域的范围及其与其他知识域的关系。主题分解构成了知识域描述的核心,知识域分解为子域、主题和子主题。对每一主题或子主题都有简短叙述,并附有一篇或不止一篇参考文献。

选择参考文献主要是因认为其构成了主题相关知识的最佳表述,并考虑到挑选参考文献的限制因素。主题和参考文献之间用对照表联系。

知识域描述的最后部分是推荐的参考文献清单,包括为希望了解知识域主题更多的用户提供的进一步阅读建议,并列出了与知识域关系最密切的标准的清单。注意,正文中方括号“[]”内的引用为推荐参考文献,圆括号“()”内的引用是用于编写和证明正文的常用参考文献。

软件需求知识域(见图 1, a 列)

软件需求是为解决某一真实问题而应展示的性质。

第一个知识子域是“软件需求基础”。包括对软件需求本身的以及对主要需求类型(产品的与过程的、功能的与非功能的、突现的性质)的定义。此子域还描述了可量化需求的重要性,并对系统需求与软件需求作了区分。

第二个知识子域是引入过程本身的“需求过程”,其中面向其余五个子域并指明需求过程如何与其他软件工程过程相衔接。该子域描述了过程模型、过程参与者、过程支持与管理及过程质量与改进。

第三个知识子域是“需求引出”,关注于软件需求出自何处及软件工程师如何收集这些需求。该子域包括需求来源及引出技术。

第四个知识子域“需求分析”关注于分析需求过程,以便:

- 检测 and 解决各需求间的冲突;
- 发现软件的界限以及软件如何与其环境进行必要的交互;
- 将系统需求详述为软件需求。

需求分析包括需求分类、概念建模、体系结构设计及需求分配及需求协商。

第五个知识子域是“需求规格说明”,通常指产生文档或等同的电子版本,能系统地评审、评价以及审批。对复杂系统,特别是涉及大量非软件部件的,就要产生多达三种不同类型的文档,包括系统界定文档、系统需求规格说明及软件需求规格说明。此子域描述这三类文档及基本活动。

第六个知识子域是“需求确认”,目的是在提交资源以处理需求之前,找出所有的问题。需求确认涉及检查需求文档的过程,以确保这些需求界定了正确的系统(即用户期望的系统)。该子域分别描述以下主题:需求评审、原型制作、模型确认及验收测试。

第七个知识子域是“实践考虑”,描述需要在实践中理解的主题。第一个主题是需求过程的迭代本质。其后三个主题主要涉及变更管理以及处于准确反映构建中软件或已构建软件状态的需求维护,分别包括:变更管理,需求属性,及需求追踪。最后一个主题是需求测量。

软件设计知识域(见图 1, b 列)

根据 IEEE 的定义[IEEE 610.12-90],软件设计既是“界定系统或部件的体系结构、部件、接口和其

他特性的过程”，又是“此过程的结果”。本知识域分为六个子域。

第一个子域提出“软件设计基础”，形成理解软件设计作用和范围的初步基础，即广义设计概念、软件设计语境、软件设计过程和使能技术。

第二个子域将“软件设计的关键问题”归于一组。这些问题包括并发性、事件控制与处理、部件分配、错误及异常处理与故障容耐、交互与表示及数据持久性。

第三个子域是“软件结构与体系结构”，其主题有总体结构与视点、体系结构风格、设计模式，最后是程序族与框架族。

第四个子域描述“软件设计质量分析与评价”。尽管整个知识域都专注于软件质量，但此子域提出了与软件设计特别相关的主题，即质量属性、质量分析与评价技术及测度。

第五个子域是“软件设计记法”，分为结构描述和行为描述。

第六个子域描述“软件设计策略与方法”。首先描述通用策略，其次依次描述面向功能的(结构化)设计方法、面向对象的设计方法、数据结构集中式设计方法及基于部件的设计(CBD)方法。

软件构造知识域(见图 1, c 列)

软件构造指通过编码、验证、单元测试、综合测试、调试，精细地创建有效用、有意义的软件。此知识域包括三个子域。

第一个子域是“软件构造基础”。前三个主题是构造的基本原理，即复杂性最小化、对变更的预测及易于验证的构造。最后的主题论述构造的相关标准。

第二个子域描述“构造管理”。其主题是各种构造模型、构造规划和构造测量。

第三个子域描述“实践考虑”。其主题是构造设计、构造语言、编码、构造测试、重用、构造质量及集成。

软件测试(见图 1, d 列)

软件测试是指以有限的测试用例对程序行为进行的动态验证。测试用例从通常无限的执行域中，针对期望的行为适当选出。软件测试有五个子域。

第一个子域是对“软件测试基础”的描述。首先提出测试相关术语，其次描述测试的关键问题，最后是测试与其他活动的联系。

第二个子域是“测试级别”。级别根据测试对象和测试目的来划分。

第三个子域是“测试技术”。第一类是基于测试者直觉和经验的技术。第二类包括：基于规格说明的技术，基于代码的技术，基于故障的技术，基于用法的技术及有关应用性质的技术。本子域还提出如何选择和组合适当的技术。

第四个子域覆盖“测试相关测度”。这些测度分为对在测程序的评价和对已完成测试的评价两类。

最后一个子域描述“测试过程”，包括实践考虑和测试活动。

软件维护(见图 1, e 列)

软件运行中一旦暴露出反常情况，操作环境立即改变，用户新的需求随之浮现。虽然软件生存周期的维护阶段从交付起就算开始，但维护活动远先于此发生。软件维护知识域分为四个子域。

第一个子域提出“软件维护基础”：术语、维护的本质、维护的必要性、维护的主要成本、软件的演化及维护。

第二个子域汇集“软件维护的关键问题”。包括技术问题、管理问题、维护成本估算及软件维护测量。

第三个子域描述“维护过程”。其主题为各种维护过程及维护活动。

第四个子域是“维护技术”，包括程序理解、重构工程和逆向工程。

软件配置管理(见图 1,f 列)

软件配置管理(SCM)是在系统整个生存周期内,为系统地控制配置变更并维护配置的完整性和可追踪性,而在不同时间点上标识软件配置的学科。此知识域包括六个子域。

第一个子域是 SCM 过程管理。覆盖的主题有 SCM 组织语境、SCM 过程约束与指南、SCM 规划、SCM 计划及 SCM 监督。

第二个子域是“软件配置标识”,包括标识受控项、建立受控项及其版本的标识方案,以及确定用于获取和管理受控项的工具和技术。此子域首要主题是受控项标识及软件库。

第三个子域是“软件配置控制”,是在软件生存周期内对变更的管理。三个主题是软件变更的请求、评价与审批以及软件变更的实现、偏离与豁免。

第四个子域是“软件配置状态记述”。主题是软件配置状态信息及软件配置状态报告。

第五个子域是“软件配置审核”,包括软件功能配置审核、软件物理配置审核、软件基线过程内审核。

最后一个子域是“软件发布管理与交付”,覆盖软件构建及软件发布管理。

软件工程管理(见图 1,g 列)

软件工程管理知识域处理对软件工程的管理和测量。尽管测量对所有知识域都很重要,然而此处是提出测量程序这一主题。共有六个子域,前五个子域覆盖软件项目管理,第六个子域描述软件测量程序。

第一个子域是“初启与范围界定”,包括需求确定与协商、可行性分析及需求评审修订过程。

第二个子域是“软件项目规划”,包括过程规划、可交付件确定、工作量、调度与成本估算、资源分配、风险管理、质量管理及计划管理。

第三个子域是“软件项目实施”。主题有计划实现、供方合同管理、测量过程实现、监督过程、控制过程及报告。

第四个子域是“评审与评价”,主题包括需求满意度确定、效能评审与评价。

第五个子域描述“闭合”,包括闭合确定、闭合活动。

第六个子域描述“软件工程测量”,更专门的是测量程序。在“软件工程过程知识域”中对产品和过程的测量进行了描述。其他很多知识域也描述了各自知识域专用的测量。本子域的主题包括测量承诺的建立与维持、测量过程规划、测量过程执行及测量评价。

软件工程过程(见图 1,h 列)

软件工程过程知识域关注于软件工程过程本身的界定、实现、评估、测量、管理、变更及改进。分四个子域。

第一个子域提出“过程实现与变更”。其主题有过程基础设施、软件过程管理周期、过程实现与变更模型及实践考虑。

第二个子域涉及“过程界定”。主题包括软件生存周期模型、软件生存周期过程、过程界定记法、过程适应及自动化。

第三个子域是“过程评估”。主题包括过程评估模型、过程评估方法。

第四个子域描述“过程与产品测量”。软件工程过程覆盖一般产品测量及一般过程测量。各知识域专用的测量在相关知识域中描述。本子域的主题有过程测量、软件产品测量、测量结果的质量、软件信息模型及过程测量技术。

软件工程工具与方法(见图 1,i 列)

软件工程工具与方法知识域包括软件工程工具和软件工程方法两方面。

软件工程工具子域采用与本指导性技术文件相同的结构,为其他九个软件工程知识域各提供一个主题。还提供一个附加主题:其他工具问题,例如工具集成技术,具有适用于所有工具类的潜力。

软件工程方法子域分为三个子域:处理非形式途径的启发式方法,处理基于数学途径的形式方法,以及处理基于各种形式的原型制作的软件开发途径的原型制作方法。

软件质量(见图 1,j 列)

软件质量知识域涉及超越软件生存周期过程的软件质量考虑。既然软件质量在软件工程中是无不关心的问题,在许多其他知识域中就也是要考虑的,且对这些知识域的指引贯穿于本知识域。本知识域的描述覆盖三个子域。

第一个子域描述“软件质量基础”,包括软件工程文化与道德、质量的价值与成本、模型与质量特性及质量改进。

第二个子域覆盖“软件质量管理过程”。主题有软件质量保证、验证和确认及评审与审核。

第三个子域描述关于软件质量的“实践考虑”。主题有软件质量需求、缺陷表征、软件质量管理技术及软件质量测量。

软件工程相关的学科(见图 1,k 列)

最后一章标题为“软件工程相关学科”。为了给软件工程划定范围,有必要认定与软件工程有公共边界的学科。

软件工程 软件工程知识体系指南

1 范围

本指导性技术文件描述了软件工程学科的边界范围,按主题提供了访问支持该学科的文献的途径。制定软件工程知识体系(SWEBOK)指南有以下五个目标:

- a) 促进业界对软件工程看法趋于一致;
- b) 阐明软件工程的地位,并设定软件工程与计算机科学、项目管理、计算机工程和数学等其他学科之间的界线;
- c) 描述软件工程学科的内容;
- d) 提供使用软件工程知识体系的主题;
- e) 为课程制定、个人认证及特许资料提供依据。

2 软件需求

2.1 综述

2.1.1 缩略语

下列缩略语适用于本文件。

DAG:有向无圈图(Directed Acyclic Graph)

FSM:功能规模测量(Functional Size Measurement)

INCOSE:国际系统工程理事会(International Council on Systems Engineering)

SADT:结构化分析与设计技术(Structured Analysis and Design Technique)

UML:统一建模语言(Unified Modeling Language)

2.1.2 导引

软件需求知识域(KA)涉及软件需求的引出、分析、规格说明及确认。软件业界公认,当这些活动不能顺利开展时,软件工程项目就会极为脆弱。

对用于解决某一现实问题的软件产品,软件需求表述了要求和约束[Kot00]。

“需求工程”一语广泛用于系统地处理需求的领域。不过为一致性起见,本指导性技术文件中将不使用该术语,原因是已经决定除软件工程的以外,避免对活动使用“工程”一语。

出于同样的原因,本指导性技术文件也不使用一些文献中出现的术语“需求工程师”,而使用术语“软件工程师”,或在某些特定情况使用“需求专家”。“需求专家”这一有争议的角色通常由软件工程师以外的个人承担,但这并暗指软件工程师不能承担这一职能。

知识域的分解结构与 ISO/IEC 12207 中关于需求活动的章节大体兼容。

提出的分解结构有一种内在风险:可推断为瀑布似的过程。为预防该风险,设计了第二个子域“需求过程”,以便通过设置过程运行所需的资源和约束以及说明采取什么措施进行设置,来提供需求过程的高层概观。

替代的分解方式可采用基于产品的结构(系统需求、软件需求、原型、用况等)。基于过程的分解反映出的事实是:需求过程要成功时,就应视为一种涉及复杂的、紧密耦合(时序及并发)活动的过程,而不