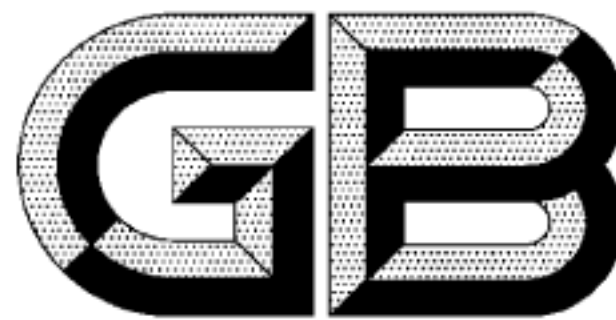




中华人民共和国国家标准

GB/T 25000.23—2019

代替 GB/T 16260.2—2006, GB/T 16260.3—2006

系统与软件工程 系统与软件质量要求和评价(SQuaRE) 第 23 部分:系统与软件产品质量测量

Systems and software engineering—
Systems and software Quality Requirements and Evaluation(SQuaRE)—
Part 23: Measurement of system and software product quality

[ISO/IEC 25023:2016, Systems and software engineering—
Systems and software Quality Requirements and Evaluation(SQuaRE)—
Measurement of system and software product quality, MOD]

2019-08-30 发布

2020-03-01 实施

国家市场监督管理总局 发布
中国国家标准化管理委员会

目 次

前言 III

引言 V

1 范围 1

2 符合性 1

3 规范性引用文件 1

4 术语和定义 2

5 缩略语 3

6 系统与软件产品质量测度的使用 3

 6.1 系统/软件产品质量测量概念 3

 6.2 质量测量的方法 4

7 质量测度的记录格式 6

8 系统与软件产品质量测度 6

 8.1 概述 6

 8.2 功能性测度 6

 8.3 性能效率测度 8

 8.4 兼容性测度 12

 8.5 易用性测度 14

 8.6 可靠性测度 21

 8.7 信息安全性测度 24

 8.8 维护性测度 28

 8.9 可移植性测度 32

附录 A（资料性附录） 本部分与 ISO/IEC 25023:2016 的结构性差异 35

附录 B（资料性附录） 质量测度使用时的考虑 39

附录 C（资料性附录） 利用质量测度元素定义产品或系统质量测度 43

附录 D（资料性附录） 测量类型的详细说明 46

参考文献 51

前 言

GB/T 25000《系统与软件工程 系统与软件质量要求和评价(SQaRE)》分为以下部分：

- 第 1 部分：SQaRE 指南；
- 第 2 部分：计划与管理；
- 第 10 部分：系统与软件质量模型；
- 第 12 部分：数据质量模型；
- 第 20 部分：测量参考模型和指南；
- 第 21 部分：质量测度元素；
- 第 22 部分：使用质量测量；
- 第 23 部分：系统与软件产品质量测量；
- 第 24 部分：数据质量测量；
- 第 30 部分：质量需求；
- 第 40 部分：评价过程；
- 第 41 部分：开发方、需方和独立评价方的评价指南；
- 第 45 部分：可恢复性的评价模块；
- 第 51 部分：就绪可用软件产品(RUSP)的质量要求和测试细则；
- 第 62 部分：易用性测试报告行业通用格式(CIF)。

本部分是 GB/T 25000 的第 23 部分。

本部分按照 GB/T 1.1—2009 给出的规则起草。

本部分代替 GB/T 16260.2—2006《软件工程 产品质量 第 2 部分：外部度量》和 GB/T 16260.3—2006《软件工程 产品质量 第 3 部分：内部度量》。本部分与 GB/T 16260.2—2006 和 GB/T 16260.3—2006 相比，主要技术变化如下：

- 将 GB/T 16260.2—2006、GB/T 16260.3—2006 中的外部度量、内部度量进行合并，更名为系统与软件产品质量测量，并将其 6 个特性的质量度量调整为 8 个特性的质量测度，修订细节如下：
 - a) 删除功能性的安全保密性度量，新增信息安全性测度，其子特性的测度包括保密性测度、完整性测度、抗抵赖性测度、可核查性测度、真实性测度和信息安全性的依从性测度；
 - b) 增加了兼容性测度，其子特性测度包括共存性测度(原可移植性的子特性测度)、互操作性测度(原功能性的子特性测度)和兼容性的依从性测度；
 - c) 增加了如下子特性的测度：功能性的功能完备性测度、性能效率的容量测度、易用性的用户差错防御性测度和易访问性测度、可靠性的可用性测度、维护性的模块化测度和可重用性测度；
 - d) 将维护性的易改变性测度和稳定性测度合并成新的测度，即易修改性测度；
 - e) 修改了特性或子特性测度的名称，新名称更准确，包括：准确性度量更名为功能正确性测度、适合性度量更名为功能适合性测度、效率度量更名为性能效率测度、效率依从性度量更名为性能效率的依从性测度、易理解性度量更名为可辨识性测度、吸引性度量更名为用户界面舒适性测度。
- 给出的质量测度表包括 ID、名称、描述、测量函数和方法，简化了 GB/T 16260.2—2006、GB/T 16260.3—2006 度量表的内容。

——在 6.1 增加了质量模型、质量测度、质量测度元素、量化属性、目标实体的关系说明。

——在合适的情况下,对特性及子特性的测度宜扩展到计算机系统的范畴,而不仅限于软件范畴。

本部分使用重新起草法修改采用 ISO/IEC 25023:2016《系统与软件工程 系统与软件质量要求和评价(SQaRE) 系统与软件产品质量测量》。

本部分与 ISO/IEC 25023:2016 相比在结构上有部分调整,附录 A 中列出了本部分与 ISO/IEC 25023:2016 的章条编号对照一览。

本部分与 ISO/IEC 25023:2016 的主要技术性差异如下:

——依据 GB/T 25000.10—2016 定义的系统与软件产品质量模型,本部分每个质量特性的测度均增加了依从性测度,分别是:功能性的依从性测度、性能效率的依从性测度、兼容性的依从性测度、易用性的依从性测度、可靠性的依从性测度、信息安全性的依从性测度、维护性的依从性测度和可移植性的依从性测度;

——第 7 章增加了有关“方法”的描述;

——在本部分的每个质量测度表中,均增加了“方法”一栏,给出了质量测度中所选用的方法;

——附录 B 表 B.1 中增加了对各质量特性依从性的说明。

本部分还做了下列编辑性修改:

——为与 GB/T 25000 系列国家标准相协调,标准名称中增加了“第 23 部分”。

——调整了参考文献顺序。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

本部分由全国信息技术标准化技术委员会(SAC/TC 28)提出并归口。

本部分起草单位:上海计算机软件技术开发中心、中国电子技术标准化研究院、中国司法大数据研究院有限公司、中国合格评定国家认可中心、河北省软件评测中心、国家应用软件产品质量监督检验中心、广西达译商务服务有限公司、北京梆梆安全科技有限公司、南京大学、中电莱斯信息系统有限公司、山东道普测评技术有限公司、中国电子科技集团公司第五十四研究所、厦门理工学院。

本部分主要起草人:蔡立志、张旻旻、王珩、赵毅、刘潇健、郭新伟、李彦军、邓姿娴、吴克寿、胡芸、龚家瑜、刘振宇、孙继欣、葛建新、周悦、王威、黄家裕、秦泗强、孙纪敏、卢俊文、崔建峰、陈振宇、房春荣、周晓明、韩庆良、韩明军、李浩文、康京山、郭晓珍、秦晓秋、韦晓枝、庞懿丽、卢梁春。

本部分所代替标准的历次版本发布情况为:

——GB/T 16260.2—2006;

——GB/T 16260.3—2006。

引 言

GB/T 25000 的本部分提供了一组用于系统/软件产品特性的质量测度,这些测度可以联合系统与软件质量要求和评价 (SQuaRE) 的其他标准使用,尤其是与 GB/T 25000.10、GB/T 25000.30、GB/T 25000.40及 GB/T 25000.41,用于规定系统/软件产品质量需求、测量和评价。开发方、评价方、质量管理方、需方、供方、维护方和目标系统/软件产品的用户可以根据需要选择对应的质量测度。这可用于定义需求,评价系统/软件产品,执行质量管理活动或者其他一些目的。

本部分中的质量测度是基于其在实践中的价值而被选入的,并将它们分成通用(G)和特定(S)两类。这些质量测度并非是详尽的,鼓励本部分的用户在必要时对它们加以细化。

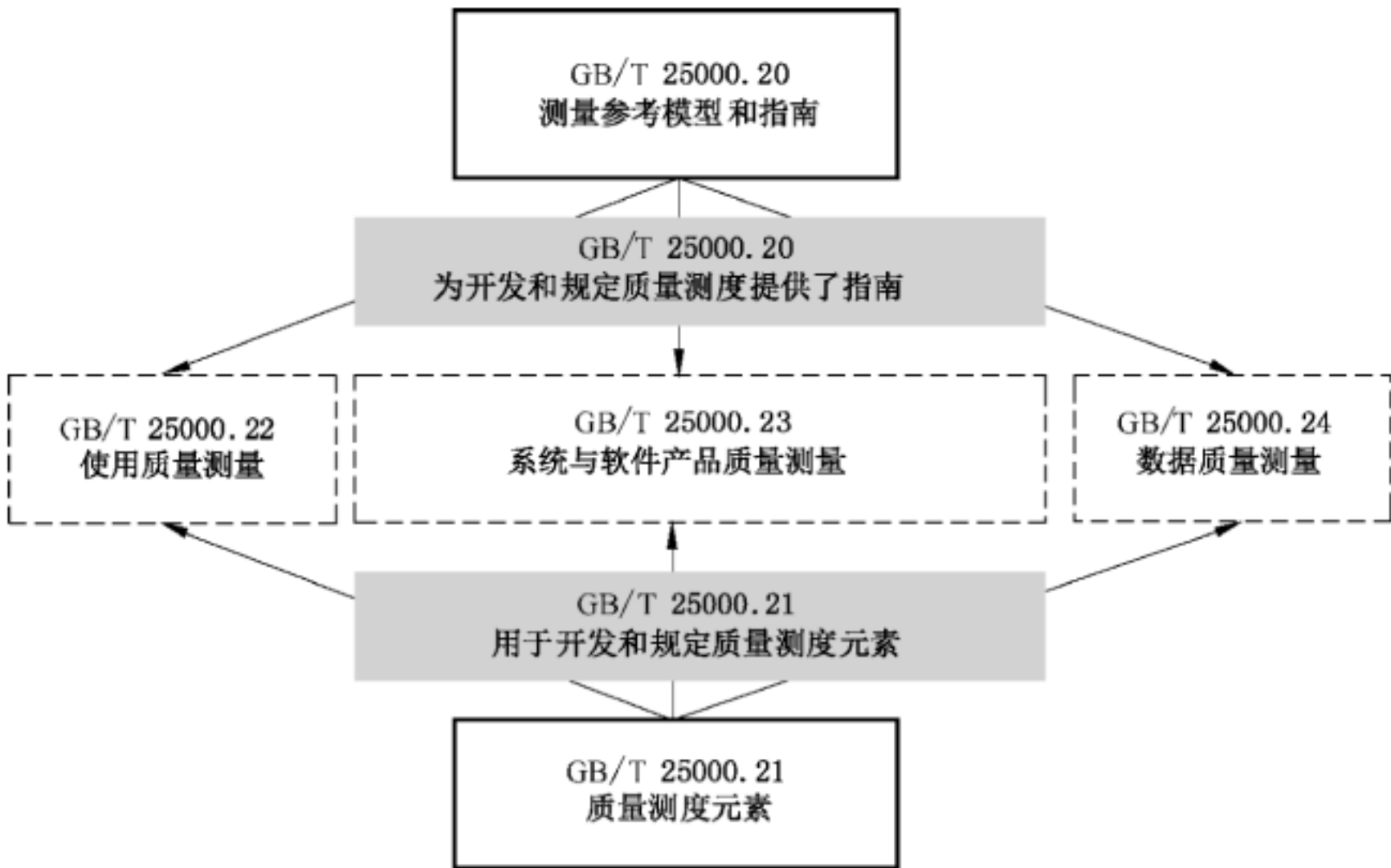


图 1 质量测量分部的结构

图 1 描绘了 GB/T 25000.23 与其他 SQuaRE 质量测量标准的关系：

- a) GB/T 25000.20——测量参考模型和指南：提供用于测量质量特性的参考模型和指南。
- b) GB/T 25000.21——质量测度元素：提供用于规定质量测度元素的一种格式,以及能够用于构建软件质量测度的质量测度元素的若干示例。
- c) GB/T 25000.22——使用质量测量：提供测度,包括使用质量模型中的质量特性的关联测量函数。
- d) GB/T 25000.23——系统与软件产品质量测量：提供测度,包括产品质量模型中的质量特性的关联测量函数和质量测度元素。
- e) GB/T 25000.24——数据质量测量：提供测度,包括数据质量模型中的质量特性的关联测量函数和质量测度元素。

系统与软件工程

系统与软件质量要求和评价(SQuaRE)

第 23 部分:系统与软件产品质量测量

1 范围

GB/T 25000 的本部分基于 GB/T 25000.10—2016 定义的特性和子特性,规定了用于量化评价系统与软件产品质量的测度。本部分定义的质量测度需要与 GB/T 25000.10—2016 协同使用,并可以联合系统与软件质量要求和评价(SQuaRE)系列国际标准的质量需求分部(ISO/IEC 2503n)及评价分部(ISO/IEC 2504n),以便更广泛地满足用户对于软件产品和系统质量需求的定义与评价。

本部分包含以下内容:

- 每个特性和子特性的质量测度的基本集;
- 如何应用软件产品和系统质量测度的解释。

本部分在附录中给出了质量测度使用时的考虑(附录 B),利用质量测度元素定义产品或系统质量测度(附录 C)以及测量类型的详细说明(附录 D)。

本部分没有为某个评定等级或依从性等级而设置测度值的范围,因为这些值是依据每个系统、产品或产品的一部分的本质而确定的,并依赖于软件类别、完整性级别和用户需求。一些属性可能会有期望的取值范围。这些范围不依赖于特定用户需求,而是依赖于通用因素,例如人的认知因素。

本部分提出的质量测度主要用于质量保证以及开发生存周期中或之后的系统与软件产品改进。

本部分的主要用户为实施质量需求规格说明和评价活动的人员,这些活动包括:

- 开发:包括软件生存周期过程中的需求分析、规格说明设计、编码、测试,直至验收活动;
- 质量管理:软件产品或计算机系统的系统性检查,例如当系统或软件产品质量评价成为质量保证、质量控制和质量认证的一部分时;
- 供应:按所签合同向需方提供系统、软件产品或软件服务时的约定,例如在合格性测试中确认软件质量;
- 获取:包括从供方获取或采购系统、软件产品或软件服务时的产品选型和验收测试;
- 维护:基于质量测量的软件产品或系统的优化。

2 符合性

遵循本部分的任何质量需求规约或质量评价应:

- a) 选择 GB/T 25000.10—2016 中定义的质量特性和/或子特性进行规定或评价;
- b) 对于所选择的每一个特性或子特性,宜使用所有第 8 章定义的通用质量测度(G),如果没有使用,应提供依据;
- c) 可选择第 8 章中定义的相关特定质量测度(S);
- d) 如果修改任何一个质量测度,应提供修改的依据;
- e) 定义额外的、本部分未包括的质量测度以及符合 GB/T 25000.21—2019 的质量测度元素。

3 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件,仅注日期的版本适用于本文

件。凡是不注日期的引用文件,其最新版本(包括所有的修改单)适用于本文件。

GB/T 25000.10—2016 系统与软件工程 系统与软件质量要求和评价(SQuaRE) 第10部分:系统与软件质量模型(ISO/IEC 25010:2011,MOD)

GB/T 25000.21—2019 系统与软件工程 系统与软件质量要求和评价(SQuaRE) 第21部分:质量测度元素(ISO/IEC 25021:2012,IDT)

ISO/IEC 25000 系统与软件工程 系统与软件质量要求和评价(SQuaRE) SQuaRE 指南[Systems and software engineering—Systems and software Quality Requirements and Evaluation (SQuaRE)—Guide to SQuaRE]

4 术语和定义

GB/T 25000.10—2016、ISO/IEC 25000 界定的以及下列术语和定义适用于本文件。

注:为便于使用,将 SQuaRE 系列标准和其他 ISO 标准中的基本定义复制于此。

4.1

(系统或软件质量的)外部测度 external measure (of system or software quality)

系统或软件产品在指定条件下使用时,其行为能满足系统(含软件)的明确和隐含要求的程度的测度。

注1:在测试和运行期间,通过执行系统或软件产品来验证和/或确认行为的属性。

注2:改写 ISO/IEC 25000:2014,定义 4.11。

示例:在测试期间发现的测试用例的失效密度是一种与计算机系统可靠性有关的软件质量外部测度。

4.2

(软件质量的)内部测度 internal measure (of software quality)

软件产品在指定条件下使用时,其静态属性满足明确和隐含要求的程度的测度。

注1:静态属性包括那些与软件架构、结构及其组件有关的属性。

注2:静态属性可通过评审、审查、模拟和/或自动化工具来验证。

注3:改写 ISO/IEC 25000:2014,定义 4.16。

示例:代码行数、复杂度和在走查中发现的故障数,都是产品自身的软件内部质量测度。

4.3

作业 job

由计算机实现的用户定义的工作单元。

注1:工作单元由一组计算机操作组成。

注2:改写 ISO/IEC/IEEE 24765:2010,定义 3.1542。

4.4

测度 measure

以测量结果来赋值的变量。

注1:术语“测度”作为集合名词,用来指代基本测度、导出测度和指标。

注2:本部分中,当“测度”被用来量化质量特性或子特性时,它指的是定义 4.8 中的一个质量测度。

注3:改写 ISO/IEC 15939:2007,定义 2.15。

4.5

测量 measurement

以确定测度的值为目的的一组操作。

注1:测量可包括分配一个定性的类别,例如源程序的语言(Ada、C、COBOL 等)。

注2:改写 ISO/IEC 15939:2007,定义 2.17。

4.6

测量函数 measurement function

为组合两个或多个质量测度元素而执行的算法或计算。

[GB/T 25000.21—2019, 定义 4.7]

4.7

量化属性 property to quantify

与质量测度元素相关,并能由某一测量方法予以量化的目标实体的性质。

注 1: 软件制品是目标实体的一种示例。

注 2: 子属性与属性相关。

[GB/T 25000.21—2019, 定义 4.11]

4.8

质量测度 quality measure

至少两个质量测度元素值的测量函数的导出测度。

[GB/T 25000.21—2019, 定义 4.13]

4.9

质量测度元素 quality measure element; QME

以某一属性及其量化的测量方法(可选地包括通过数学函数所做的变换)来定义的测度。

[GB/T 25000.21—2019, 定义 4.14]

4.10

质量模型 quality model

定义的特性集以及它们之间的关系集,为规约质量需求以及评价质量提供了一个框架。

[ISO/IEC 25000:2014, 定义 4.27]

4.11

(软件产品或系统的)质量特性 quality characteristic (of software product or system)

支撑软件产品或系统质量的质量属性的类别。

注: 改写 ISO/IEC 25000:2014, 定义 4.34。

4.12

任务 task

实现目标所必需的活动集合。

注 1: 这些活动可能是物理活动或认知活动。

注 2: 角色和职责可能决定目标和任务。

注 3: 改写 ISO/IEC 9241-11:1998, 定义 3.9。

5 缩略语

下列缩略语适用于本文件。

QME: 质量测度元素(Quality Measure Element)

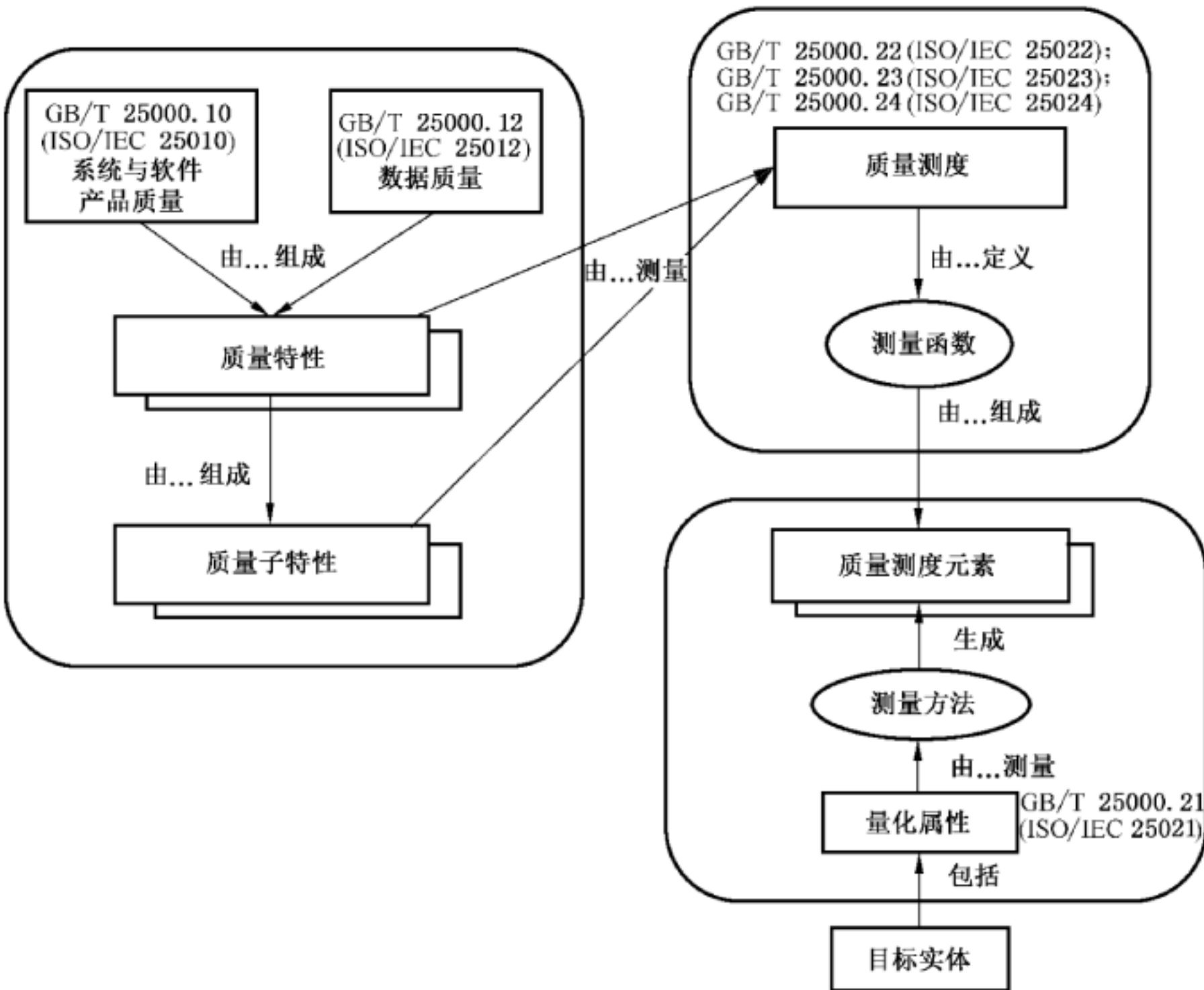
6 系统与软件产品质量测度的使用

6.1 系统/软件产品质量测量概念

系统/软件产品质量是满足各种各样利益相关方明确和隐含要求的程度并提供量值。系统/软件质量要求和评价(SQuaRE)系列标准通过质量模型表示这些明确和隐含的要求,质量模型把系统/软件产

品质量分成多个特性,在某些情况下这些特性又被分成子特性。系统/软件产品的可测量的质量相关属性被称作量化属性且与质量测度有关。这些属性通过应用测量方法进行测量。测量方法是一种逻辑操作序列,用于量化规定标度的属性。应用测量方法的结果被称作质量测度元素。

质量特性和子特性可以使用测量函数进行量化。测量函数是一个用来组合质量测度元素的算法。应用测量函数的结果被称作质量测度。从这方面看,质量测度可量化质量特性和子特性。一个质量特性或子特性可以用多个质量测度来测量(见图 2)。



注：目标实体可以是一个系统、一个软件产品、数据或者一个用户（见 GB/T 25000.10—2016，图 5）

图 2 质量模型、质量测度、质量测度元素、量化属性和目标实体间的关系

6.2 质量测量的方法

用户质量要求包括特定使用周境中的系统使用质量的需求。当使用软件产品质量特性和子特性来规定质量的外部测度和内部测度时,可以考虑这些被标识的要求。

软件产品质量可以通过测量内部属性(典型的是对中间产品的静态测度),也可以通过测量外部属性(典型的是通过测量代码执行时的行为),或者通过测量使用质量的属性(当产品在真实场景或模拟场景中使用)来评价。合适的软件内部属性是获得所需外部行为的先决条件,而适当的外部行为则是获得使用质量的先决条件(见图 3)。

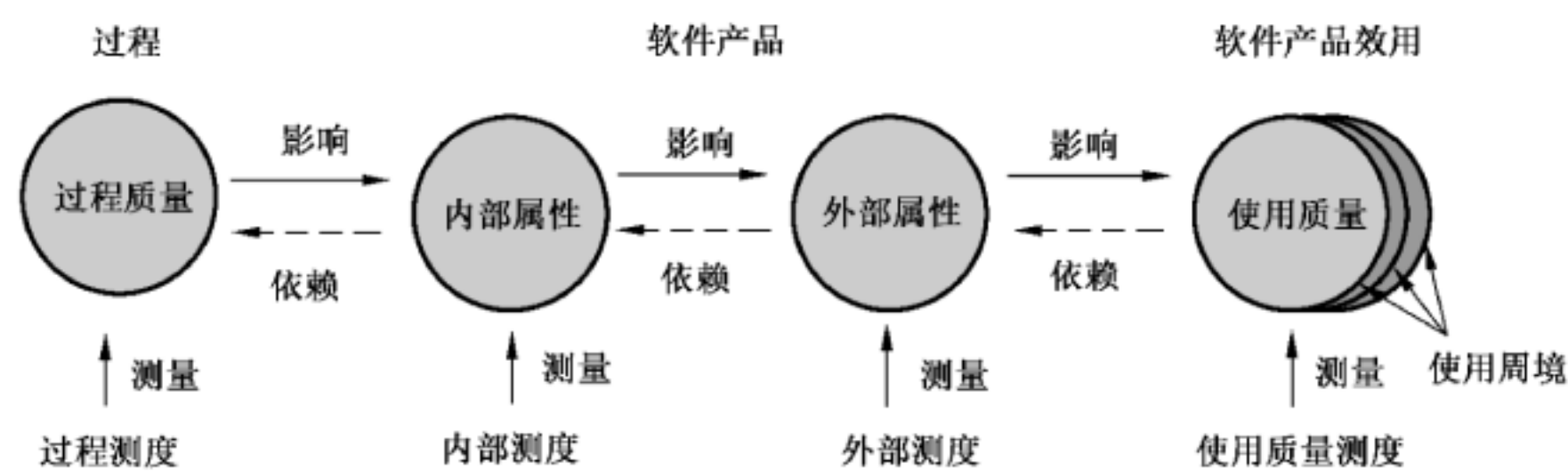


图 3 质量测量类型间的关系

内部测度可应用于开发阶段的非执行系统/软件产品(比如标书、需求定义、设计规格说明或源代码等),它可以采用评审、审查、模拟和自动化工具等方式进行验证。内部测度为用户提供了测量中间可交付项质量的能力,从而可以预测最终产品的质量,这样就可以使用户尽可能在开发生存周期早期察觉质量问题,并采取纠正措施。例如,复杂度、在走查中发现的故障数量、严重程度以及故障失效频率,是对软件产品本身进行的内部质量测度。

外部测度可以通过该系统/软件产品的系统行为来测量其质量,系统行为是系统/软件产品的一部分。外部测度只能在生存周期过程中的测试阶段和任何运行阶段使用。在测试和/或期望操作的系统环境中运行系统/软件产品,即可获得这样的测量。例如,测试过程中发现的失效数是与计算机系统中出现的故障数有关的一个软件质量外部测度。

建议尽可能采用与目标外部测度有密切关系的内部测度,以便使用这些内部测度来预测外部测度的值。

本部分提供了一组建议的系统与软件质量测度(内部测度和外部测度),与 GB/T 25000.10—2016 的质量模型联合使用。本部分的用户可以修改已定义的质量测度,也可以定义、使用尚未在本部分中定义的质量测度。

注 1: 例如,质量特性的特定测量,如安全性或信息安全性,可以参见由 IEC 65 和 ISO/IEC JTC 1/SC 27 提供的标准。

当修改使用或使用尚未列在本部分中的新的质量测度时,用户应说明该测度与 GB/T 25000.10—2016 质量模型或其他可替换质量模型的相关性。

大部分质量测度采用一个测量函数,该函数将结果归一化到 0.0 到 1.0 的区间范围,且越接近 1.0 越好。如果不是这种情况,其解释以注的形式给出。一些质量测度产生的结果与目标值相关,该目标值需要作为需求的一部分来建立。

注 2: 有些测量是针对需求规格说明、设计规格说明或用户文档中规定的目标值进行归一化。开发方或维护方通过改进体系结构、设计、实现、装配、操作规程、用户接口或软件产品或系统的性能,可以确定这些目标值,并作为阈值。目标值也可以作为需方和供方商定的需求之一,用来规定质量需求或检查获取的符合性。开发过程中需求规格说明经常变动和修改,这会影响基于它的质量测度。在开发初期,很难完全明确利益相关方或系统需求产生的明确和隐含要求。一些要规定的需求可能遗漏或不一致,或者一些目标值可能不充分,并需要改变。因此,质量测度的用户需要考虑改进和修订需求规格说明,并在开发和/或评价阶段反复应用质量测度。

注 3: 一些质量测度(例如平均响应时间)可能难以单独解释,下面列举了几种质量测度应用的方法,以便理解与解释:

- a) 符合性:将测度与特定业务或使用需求进行比较(例如:最大可接受的响应时间是 0.5 s);
- b) 基准:为了相同目的,将测度与相同或相似产品或系统的基准进行比较(例如:新系统的平均响应时间不超过老系统的平均响应时间);
- c) 时序:比较随时间变化的趋势(例如一天中平均响应时间的变化)。

7 质量测度的记录格式

第 8 章质量测度表格中各字段的含义如下：

- a) ID:质量测度编号,每个 ID 包含三个部分的内容:
 - 缩略字母:质量特性表示为大写字母 X,子特性表示为大写字母 X 与小写字母 x 的组合(例如,“PTb”表示性能效率中的时间特性测度);
 - 质量子特性内的序号则采用阿拉伯数字表示;
 - G(通用)或 S(特定)表示质量测度的潜在类别;G(通用)表示在合适时都可以使用,S(特定)表示在特定的场景中使用。
 - b) 名称:质量测度的名称。
 - c) 描述:质量测度的信息。
 - d) 测量函数:将质量测度元素结合起来产生质量测度的数学公式。
- 注:附录 C 简要给出了用于构建质量测度的常用的质量测度元素的说明,以帮助理解和应用质量测度的测量函数。
- e) 方法:可用于获取质量测度的方法,在具体操作中方法不唯一。

8 系统与软件产品质量测度

8.1 概述

第 8 章的质量测度按照 GB/T 25000.10—2016 中的质量特性和子特性给出。

质量测度可根据质量特性以及评价等级选择不同的评价技术,这取决于它是作为内部测度还是外部测度。因此,第 8 章列出的一些质量测度可以用在不同的评价阶段,例如设计规格说明的静态评审或可执行产品的动态分析。

可应用的质量测度并不局限于本章所列举的测度。推荐使用有关标准或指南中规定的测度或测量。例如,功能规模测量的定义参见 ISO/IEC 14143,精确时间效率测量的实例参见 ISO/IEC 14756。

注 1:质量测度列表是可以修改的,本部分的未来版本可能进行修订。希望本部分的读者积极提供反馈。

注 2:本章中除了特别规定外,术语“测度”均指质量测度。例如,“功能适合性测度”指“功能适合性质量测度”。

8.2 功能性测度

8.2.1 功能性测度概述

功能性测度用于评估产品或系统在指定情况下使用时,提供满足明确和隐含要求的功能的程度。

注 1:功能性关注功能是否满足明确及隐含的要求。

注 2:这里所涉及的功能可以是 ISO/IEC 14143 中定义的功能用户需求的基本过程。

注 3:采用其他质量测度元素(例如功能规模)构成的类似测度,可以作为一种更精确的对结果加权的方法,因为单位比例并不表明功能性缺失的量化程度。

8.2.2 功能完备性测度

功能完备性测度用于评估功能集对所有指定的任务或用户目标的覆盖程度(见表 1)。

表 1 功能完备性测度

ID	名称	描述	测量函数	方法
FCp-1-G	功能覆盖率	所指定功能的实现比例是多少？	$X=1-A/B$ A = 缺少的功能数量 B = 指定的功能数量	对在评价中检测到缺少的功能数进行计数,将其与需求规格说明或其他相关文档中指定的功能数相比较
注 1: 功能可以在需求规格说明、设计规格说明、用户手册中进行指定。 注 2: 当系统或软件产品无法实现指定的功能时,即检测到缺少的功能。				

8.2.3 功能正确性测度

功能正确性测度用于评估产品或系统提供具有所需精度的正确结果的程度(见表 2)。

表 2 功能正确性测度

ID	名称	描述	测量函数	方法
FCr-1-G	功能正确性	可提供正确结果的功能比例是多少？	$X=1-A/B$ A = 功能不正确的数量 B = 考虑的功能数量	选定所有功能或一组特定功能集(一般按照重要性和目的)对系统做功能测试。对不能正确实现的功能数进行计数,将其与考虑的功能总数相比较
注 1: 不正确的功能是指功能无法给出达到特定预期目标的合理、可接受的结果。 注 2: 用于评价的功能可能是产品的所有功能或者特定阶段所需要的一组特定功能集。 注 3: 开发方或维护方可能通过评审或测试检查单个功能,并决定功能能否为需求规格说明中定义的特定目标提供正确的结果。在这种情况下,正确性的程度是由每个功能决定。				

8.2.4 功能适合性测度

功能适合性测度用于评估功能促使指定的任务和目标实现的程度(见表 3)。

表 3 功能适合性测度

ID	名称	描述	测量函数	方法
FAp-1-G	使用目标的功能适合性	用户要求的功能中提供了合适结果以实现特定使用目标的比例是多少？	$X=1-A/B$ A = 为实现特定使用目标所需的功能中缺少或不正确功能的数量 B = 为实现特定使用目标所需的功能数量	依据需求规格说明、操作手册等,针对为实现特定使用目标所需的功能进行测试,分析测试结果,确定缺少或不正确的功能数量,并与要求实现特定使用目标所需的功能数进行比较
注 1: 这个测量函数通常用在最重要的或频繁标识的使用目标上,因此首先针对系统中可获取的每个定义的使用目标计算质量测度,然后进行下一个质量测度,例如 FAp-2-G“系统的功能适合性”可以通过所有的使用目标共同计算来提供一个系统质量测度。 注 2: 本部分的用户也可以考虑测量可实现的用户目标的比例,以便更好地了解用户预期使用的实际影响。				

表 3（续）

ID	名称	描述	测量函数	方法
FAP-2-G	系统的功能适合性	用户要求达到的目标功能中,能给出合适结果的功能比例是多少?	$X = \sum_{i=1}^n A_i / n$ A_i = 使用目标 i 的适合性得分,即第 i 个特定使用目标 FAP-1-G 的测量值 n = 使用目标的数量	计算 FAP-1-G 中测量值的平均值

8.2.5 功能性的依从性测度

功能性的依从性测度用于评估产品或系统遵循与功能性相关的标准、约定或法规以及类似规定的程度(见表 4)。

表 4 功能性的依从性测度

ID	名称	描述	测量函数	方法
FCI-1-G	功能性的依从性	遵循与产品或系统的功能性相关的标准、约定或法规以及类似规定的程度如何?	$X = A / B$ A = 在评价中已证实的正确实现的与功能性的依从性相关的项数 B = 与功能性的依从性相关的项数	依据与功能性相关的依从性标准、约定或法规以及类似规定,对已满足所要求的依从性(例如系统涵盖相关规定要求的基本功能)的项数进行计数,并与需求规格说明或其他相关文档中要求的功能性的依从性项数相比较

8.3 性能效率测度

8.3.1 性能效率测度概述

性能效率测度用于评估在指定条件下使用的资源数量的性能。资源可包括其他软件产品、系统的软件和硬件配置、以及耗材(例如打印纸、存储媒体)。

注 1: 使用条件对性能效率测度有很大影响并会造成波动,例如处理数据的负载、使用频率、相连的站点数等。因此性能效率测度可以包括带误差波动的估计或测量值与规格说明中要求的允许误差波动范围的设计值的比率。建议列出并研究各种因素所起的作用,例如其他软件所用的“CPU”和内存、网络流量和定时的后台处理等。宜为估计或测量值建立合适的波动和有效范围,并把它们与需求规格说明进行比较。

注 2: 建议标识和定义适合于性能效率或容量测度的任务,例如作为业务应用任务的一个事务、作为通信应用任务的一个交换或数据包的发送、作为控制应用任务的一个事件控制,以及作为普通用户应用的由用户可调用功能产生的一个数据输出。

8.3.2 时间特性测度

时间特性测度用于评估产品或系统执行其功能时,其响应时间、处理时间及吞吐率满足需求的程度(见表 5)。

表5 时间特性测度

ID	名称	描述	测量函数	方法
PTb-1-G	平均响应时间	系统响应一个用户任务或系统任务的平均时间是多少?	$X = \sum_{i=1}^n (A_i) / n$ $A_i =$ 第 i 次测量时系统响应一个特定用户任务或系统任务花费的时间 $n =$ 测得的响应次数	测量为完成特定用户任务或系统任务而耗费的时间,并进行多次测量,计算平均时间
PTb-2-G	响应时间的充分性	系统响应时间满足规定目标的程度如何?	$X = A / B$ $A =$ PTb-1-G 测度中所测量的平均响应时间 $B =$ 规定的任务响应时间	PTb-1-G 测度中所测量的平均响应时间与用户文档(例如需求规格说明、合同等)规定的响应时间进行比较
注 1: 测度结果值越小越好,结果值小于或等于 1 为良好。 注 2: 响应时间是从提交请求开始到第一次产生响应为止所需要的时间,即这个时间指的是开始响应,而不是输出响应的时间。 注 3: 另一种可替代的测度是在期望负载条件下第 n 个百分位响应时间。同样它也适用于单个功能或功能类。				
PTb-3-G	平均周转时间	完成一个作业或一个异步进程的平均时间是多少?	$X = \sum_{i=1}^n (B_i - A_i) / n$ $A_i =$ 作业或异步进程 i 的开始时刻 $B_i =$ 作业或异步进程 i 的完成时刻 $n =$ 测量的次数	记录给定作业或异步进程的开始及完成时刻,并进行多次测量,根据给定公式计算平均周转时间
PTb-4-G	周转时间充分性	周转时间满足规定目标的程度如何?	$X = A / B$ $A =$ PTb-3-G 测度中所测量的平均周转时间 $B =$ 规定的作业或异步进程的周转时间	PTb-3-G 测度中所测量的平均周转时间与用户文档(例如需求规格说明、合同等)规定的周转时间进行比较
注 1: 测度结果值越小越好,结果值小于或等于 1 为良好。 注 2: 以流水线(例如系统链)为例,宜考虑流水线中每个阶段所用时间,每个阶段的瓶颈都影响整体周转时间。 注 3: 使用这个测度时建议结合规定的有效载荷和/或工作负载。				

表 5（续）

ID	名称	描述	测量函数	方法
PTb-5-G	平均吞吐量	单位时间内完成作业的平均数量是多少？	$X = \sum_{i=1}^n (A_i/B_i)/n$ A_i = 第 i 次观察时间内完成的作业数量 B_i = 第 i 次观察时间的周期 n = 观察的次数	按照业务模型执行作业，测量完成这些作业所需的时间，并计算单位时间内完成的作业数
注 1：作业可能是细粒度操作例如微处理器操作，或粗粒度事务处理单元例如事务处理性能委员会（TPC）定义的事务，或更高层次的抽象例如函数。所以在不同周境中使用时，应合理解释测度结果。 注 2：平均吞吐量与吞吐量目标阈值进行比较，从而计算吞吐量的充分性。当特定条件下规定的目标阈值被当作其中一个需求时，其结果需要大于 1。				

8.3.3 资源利用性测度

资源利用性测度用于评估产品或系统执行其功能时，所使用资源数量和类型满足需求的程度（见表 6）。

表 6 资源利用性测度

ID	名称	描述	测量函数	方法
PRu-1-G	处理器平均占用率	执行一组给定的任务，处理器所需要的时间与运行时间的平均比率是多少？	$X = \sum_{i=1}^n (A_i/B_i)/n$ A_i = 第 i 次观察中，处理器执行一组给定任务所用的时间 B_i = 第 i 次观察中，执行任务的运行时间 n = 观察次数	执行一组给定的具有一定特征的任务，记录处理器所用时间以及运行时间，并进行多次测量，按照给定公式计算处理器的平均占用率
注：结果值介于 0 到 1 之间，通常来说结果值越小越好。				
PRu-2-G	内存平均占用率	执行一组给定的任务所需要的内存与可用内存的平均比率是多少？	$X = \sum_{i=1}^n (A_i/B_i)/n$ A_i = 第 i 次样本处理中执行一组给定任务所占用的实际内存大小 B_i = 第 i 次样本处理期间可用于执行任务的内存大小 n = 处理的样本数	执行一组给定的具有一定特征的任务，记录所占用内存的大小以及可用内存大小，并进行多次测量，根据给定公式计算内存平均占用率
注：结果值介于 0 到 1 之间，通常来说结果值越小越好。				

表 6（续）

ID	名称	描述	测量函数	方法
PRu-3-G	I/O 设备平均 占用率	执行一组给定的任务所占用的 I/O 设备的时间与 I/O 操作时间的平均比率是多少？	$X = \sum_{i=1}^n (A_i / B_i) / n$ A_i = 第 i 次观察中, 执行一组给定任务所占用 I/O 设备的持续时间 B_i = 第 i 次观察中, 执行任务所需 I/O 运行的持续时间 n = 观察次数	执行一组给定的具有一定特征的任务, 记录占用 I/O 设备的时间以及执行任务所需的 I/O 操作时间, 并进行多次测量, 根据给定公式计算 I/O 设备平均占用率
注 1: 结果值介于 0 到 1 之间, 通常来说结果值越小越好。 注 2: 占用时间指一个系统或一个设备实际执行任务时所用的时间周期。				
PRu-4-S	带宽占用率	执行一组给定任务时使用可用带宽的比例是多少？	$X = A / B$ A = 执行一组给定任务时测得的实际传输带宽 B = 执行一组任务时可用带宽容量	执行一组给定的具有一定特征的任务, 比较实际传输带宽与可用带宽
注 1: 如果考虑相关类型的资源是否在特定的时间周期中得到较好的利用, 例如避免中断处理, 以最大资源利用率完成规定任务, 则结果值越接近最优值越好。在这种情况下, 最优值取决于所处的环境。 注 2: 测量员宜考虑可能影响结果统计值(包含均值)的通信流量的限制(例如丢包或限流)。				

8.3.4 容量测度

容量测度用于评估产品或系统参数的最大限量满足需求的程度(见表 7)。

注 1: 容量测度可通过动态分析(例如系统的容量测试)、系统集成测试或模拟的方法进行测量。大部分情况下, 可研究静态分析、动态测试或运营在持续时间内的最大值及分布。

注 2: 理论上期望的最大值被指定为超过可能实际值的目标值。

表 7 容量测度

ID	名称	描述	测量函数	方法
PCa-1-G	事务处理容量	单位时间内处理事务的数量是多少？	$X = A / B$ A = 观察时间内完成事务的数量 B = 观察时间	在给定观察时间内, 施加足够的工作负载, 测量完成事务的数量, 并计算单位时间内处理事务的数量
注 1: 结果值介于 0 到最大值之间, 通常结果值越大越好。 注 2: 只有进行足够的工作负载测试, 该测度才有意义。 注 3: 除事务外, 也可以使用任务作为测度。				

表 7（续）

ID	名称	描述	测量函数	方法
PCa-2-G	用户访问量	某一时刻,可同时访问系统的用户数量是多少?	$X = \sum_{i=1}^n A_i / n$ $A_i = \text{第 } i \text{ 次观察中,同时访问系统的最大用户数量}$ $n = \text{观察次数}$	通过对一个逐渐增大的并发请求进行测试,并可以获得请求响应时间,分析请求响应时间的突变点,从而获得最大用户数量,并进行多次测试观察
注: 结果值介于 0 到最大值之间,通常结果值越大越好。				
PCa-3-S	用户访问增长的充分性	单位时间内可成功添加用户的数量是多少?	$X = A / B$ $A = \text{观察时间内成功增加的用户数量}$ $B = \text{观察时间}$	通过逐渐增加用户量进行测试尝试,记录用户成功添加的数量,并计算单位时间内用户添加的数量
注 1: 结果值介于 0 到最大值之间,通常结果值越大越好。 注 2: 该测度表明软件或系统具有足够容量接受大量用户访问的程度,即使用户在给定时刻快速增加,例如大量用户可以通过互联网同时访问系统或软件。				

8.3.5 性能效率的依从性测度

性能效率的依从性测度用于评估产品或系统遵循与性能效率相关的标准、约定或法规以及类似规定的程度(见表 8)。

表 8 性能效率的依从性测度

ID	名称	描述	测量函数	方法
PCI-1-G	性能效率的依从性	遵循与产品或系统效率有关的标准、约定或法规以及类似规定的程度如何?	$X = A / B$ $A = \text{在评价中已证实正确实现的与效率依从性有关的项数}$ $B = \text{与性能效率的依从性相关的项数}$	依据与性能效率有关的依从性标准、约定或法规以及类似规定,对满足所要求的依从性(例如在时间、容量方面)的项数进行计数,并与需求规格说明或其他相关文档中要求的性能效率的依从性项数相比较

8.4 兼容性测度

8.4.1 兼容性测度概述

兼容性测度用于评估在共享相同的硬件或软件环境的条件下,产品、系统或组件能够与其他产品、系统或组件交换信息和/或执行其所需的功能的程度。

8.4.2 共存性测度

共存性测度用于评估与其他产品共享通用的环境和资源的条件下,产品能够有效执行其所需的功能并且不会对其他产品造成负面影响的程度(见表 9)。

表 9 共存性测度

ID	名称	描述	测量函数	方法
CCo-1-G	与其他产品的共存性	规定的其他软件产品可以与该软件产品共享环境,而不会对质量特性或功能产生负面影响的比例是多少?	$X=A/B$ A = 与该产品可共存的其他规定的软件产品数量 B = 在运行环境中,该产品需要与其他软件产品共存的数量	选择与该软件产品需要共存的软件,验证与被测软件的共存情况(例如软件安装情况、功能执行情况、系统资源占用等)

8.4.3 互操作性测度

互操作性测度用于评估两个或多个系统、产品或组件能够交换信息并使用已交换的信息的程度(见表 10)。

表 10 互操作性测度

ID	名称	描述	测量函数	方法
CIn-1-G	数据格式可交换性	与其他软件或系统交换规定的数据格式的比例是多少?	$X=A/B$ A = 与其他软件或系统可交换数据格式的数量 B = 需要交换的数据格式数量	对在测试数据交换中证实能与其他软件或系统交换的数据格式的数目进行计数,并与数据交换的总数相比较
CIn-2-G	数据交换协议充分性	支持规定的数据交换协议的比例是多少?	$X=A/B$ A = 实际支持数据交换协议的数量 B = 规定支持的数据交换协议数量	测试系统的数据交换协议符合性,例如数据正常传输和解析。对在测试数据交换中证实能与其他软件进行数据交换的协议的数目进行计数,并与数据交换协议的总数相比较
注:有关数据质量的详细介绍,可参考 GB/T 25000.24—2017 Con-I-1。				
CIn-3-S	外部接口充分性	支持规定的外部接口(与其他软件和系统的接口)的比例是多少?	$X=A/B$ A = 有效的外部接口数量 B = 规定的外部接口数量	按照相关文档的要求(例如需求规格说明、设计说明等),统计能成功与外部通信的接口数量,并与所规定的外部接口数量进行比较

8.4.4 兼容性的依从性测度

兼容性的依从性测度用于评估产品或系统遵循与兼容性相关的标准、约定或法规以及类似规定的程度(见表 11)。

表 11 兼容性的依从性测度

ID	名称	描述	测量函数	方法
CCI-1-G	兼容性的依从性	遵循与产品或系统的兼容性有关的标准、约定或法规以及类似规定的程度如何?	$X = A / B$ A = 在评价中已证实正确实现的与兼容性的依从性有关的项数 B = 与兼容性的依从性相关的项数	依据与兼容性有关的依从性标准、约定或法规以及类似规定,对已满足所要求的依从性(例如在共存性或接口一致性方面)的项数进行计数,并与需求规格说明或其他相关文档中要求的兼容性的依从性项数相比较

8.5 易用性测度

8.5.1 易用性测度概述

易用性测度用于评估在指定使用周境中,产品或系统在有效性、效率和满意度特性方面,为了达到所指定的目标可被特定用户使用的程度。

- 注 1: 内部易用性测度用来预测该软件能够被理解、学习和操作,以及使用户感受到舒适和满意交互的程度。
- 注 2: 许多外部易用性测度通过用户尝试使用某一功能来进行测试。其结果将受用户的能力和宿主系统特性的影响。这并非是使测量无效,因为被测软件要在明确规定的条件下由代表指定用户群的样本用户来运行。(对通用产品,可使用一定范围的用户群的代表。)尽管从较小的用户群中也能得到有用的信息,但为得到可靠的结果,需要使用大量忠实用户的样本。用户进行这些测试时不宜得到任何提示或外部帮助。
- 注 3: 内部和外部的易用性测度在规定的设计规约、明确的指导方针或易用性规格说明方面与实际开发文档设计、原型或可执行的系统/软件进行比较。因此,通过考虑使用质量的特性和测度以及用户中心设计理念和人类工程学观点,引出最终用户的需求并建立易用性特定规格说明是非常重要的。例如,对最终用户来说,易用性的特定准则、模板、或检查表用于详细解释什么类型的信息容易理解是很有必要的。
- 注 4: 本部分中,易用性测度的目标实体仅限于任何系统或软件产品。特定使用周境中涉及到有效性、效率和满意度的易用性测度可参见 GB/T 25000.22—2019。
- 注 5: 易用性测度将会不可避免地产生少量主观上的结果。在用比率标度测量困难的情况下,可以根据不同情况使用顺序标度作为代替,例如 1.0 代表优秀,0.8 代表良好,0.6 代表中,0.4 代表差,0.2 代表非常差。

8.5.2 可辨识性测度

用户只能够选择适合他们预期使用的一个系统/软件产品。可辨识性质量测度用于评估用户能够辨识产品或系统是否适合他们的要求的程度(见表 12)。

- 注: 可辨识性测度能够用来评估新用户是否能够理解:
- 软件产品或系统能否适合他们的目的;
 - 它如何用于特定的任务中。

表 12 可辨识性测度

ID	名称	描述	测量函数	方法
UAp-1-G	描述的完整性	在产品描述或用户文档中描述使用场景的比例是多少?	$X = A / B$ A = 在产品描述或用户文档中所描述的使用场景数量 B = 产品的使用场景数量	对产品描述或用户文档进行评审,通过评审确定能理解的使用场景数量,并与相关文档中规定的使用场景数量进行比较
UAp-2-S	演示覆盖率	有多少比例的任务具有让用户辨识其适合性的演示能力?	$X = A / B$ A = 具有演示功能的任务的数量 B = 期望能从演示功能中获益的任务数量	对需要演示的任务进行操作,对能充分演示的任务数进行计数,并与期望演示的任务总数相比较
UAp-3-S	入口点的自描述性	网站的引导页中能说明该网站目的的比例是多少?	$X = A / B$ A = 能说明网站目的的引导页数量 B = 网站中引导页的数量	对网站引导页的用途描述进行评审,确定能说明网站目的的引导页数量,并与网站中引导页的数量进行比较

8.5.3 易学性测度

易学性测度用于评估在指定使用周境中,产品或系统在有效性、效率、抗风险和满意度特性方面,为了学习使用该产品或系统这一指定目标,可为指定用户使用的程度(见表 13)。

表 13 易学性测度

ID	名称	描述	测量函数	方法
ULe-1-G	用户指导完整性	在用户文档和/或帮助机制中有多少比例的功能,能充分描述并能使用户使用这些功能?	$X = A / B$ A = 在用户文档和/或帮助机制中按要求描述的功能数量 B = 要求实现的功能总数量	对用户文档和/或帮助机制中的功能描述进行评审,并对按要求描述的功能进行计数
注 1: 易学性与可辨识性密切相关,可辨识性测量可作为软件易学性能力的指标。 注 2: 帮助机制包括,例如在线帮助、操作指导视频、操作指令系统等。 注 3: 当人类工效学或用户经验的观点得到很好地应用时,可以形成更复杂的测度,例如系统概念模型到用户心理模型映射的匹配程度。				
ULe-2-S	输入字段的默认值	在具有默认值的输入字段中,可以自动填充默认值的输入字段比例是多少?	$X = A / B$ A = 运行过程中自动填充默认值的输入字段数量 B = 具有默认值的输入字段的数量	对软件运行过程中可自动填充默认值的字段进行计数,并与要求具有默认值的输入字段数量进行比较
注: 输入字段的默认值有助于初学者全面、快速地学习如何操作产品。				

表 13 (续)

ID	名称	描述	测量函数	方法
ULe-3-S	差错信息的易理解性	差错信息中能够给出差错发生的原因以及解决方法的比例是多少?	$X=A/B$ A=给出差错发生原因及可能解决方法的差错信息数量 B=差错信息的数量	统计成功给出差错原因及解决方法的差错信息数量,并与测试过程中出现的所有差错信息的数量进行比较
ULe-4-S	用户界面的自解释性	呈现给用户的信息元素和步骤中有多大比例能帮助新用户在没有先前学习或训练或寻求外部帮助的情况下完成常规任务?	$X=A/B$ A=以用户可以理解的方式所呈现信息元素和步骤的数量 B=对于新用户来说完成常规任务所需信息元素和步骤的数量	记录以用户可理解的方式呈现的信息元素和步骤的数量,并与所需的信息元素与步骤总数相比较
注:该测度尤其与公共系统和网站相关。				

8.5.4 易操作性测度

易操作性测度用于评估产品或系统具有易于操作和控制的属性的程度(见表 14)。

注 1: 易操作性测度是通过对操作员代表或最终用户代表的操作性测试进行测量,或者通过静态分析例如对需求、设计规格说明或用户手册进行评审来测量。

注 2: 内部或外部易操作性质量测度用于评估用户能否操作和控制软件。易操作性测度可按 ISO 9241-110 中的对话原则进行分类:

- 软件对任务的适合性;
- 软件自我描述性;
- 软件的可控制性;
- 软件对用户期望的符合性;
- 软件的容错性;
- 软件对各种特例的适合性。

表 14 易操作性测度

ID	名称	描述	测量函数	方法
UOp-1-G	操作一致性	交互式任务在多大程度上具有在任务内和类似任务中一致的行为和外观?	$X=1-A/B$ A=不一致的特定交互式任务数量 B=需要一致的交互任务的数量	通过评审交互式任务的相类似的外观和操作,确定操作结果及外观具有一致性的任务数量,并与所规定的交互式任务的数量进行比较

表 14 (续)

ID	名称	描述	测量函数	方法
UOp-2-G	消息的明确性	系统能给用户传达正确结果或指令消息的比例是多少?	$X = A / B$ $A =$ 传达给用户正确结果或指令的消息数量 $B =$ 实现的消息数量	通过对消息进行评审,确定传达给用户正确结果或指令的消息数量,并与实现的消息数量进行比较
注: 消息提供可能帮助用户的所有可用信息,并在可能时说明如何解决错误。				
UOp-3-S	功能的易定制性	为使用方便,用户能够定制功能和操作规程的比例是多少?	$X = A / B$ $A =$ 为用户使用方便而提供的可被定制的功能和操作规程的数量 $B =$ 用户能够受益于定制的功能和操作规程的数量	记录可以用于用户定制的功能和操作规程的数量,并与相关文档中(例如需求规格说明、合同等)规定的可受益于定制的功能和操作规程数量进行比较
UOp-4-S	用户界面的易定制性	在外观上可以定制的用户界面元素的比例是多少?	$X = A / B$ $A =$ 可以定制的用户界面元素数量 $B =$ 期望能够受益于定制的用户界面元素数量	记录可以用于用户改变界面元素的数量,并与相关文档(例如需求规格说明、合同等)中规定的期望受益于定制用户界面元素数量进行比较
UOp-5-S	监视能力	在运营过程中,功能状态可被监视的比例是多少?	$X = A / B$ $A =$ 具有状态监视能力的功能数量 $B =$ 期望受益于监视能力的功能数量	通过执行要求具有状态监视能力的功能,并与相关文档(例如需求规格说明、合同等)中规定的期望受益于监视能力的功能数进行比较
注 1: 对于一些功能操作状态的监视和管理,例如在处理分布式系统、嵌入式系统 etc 情况下,非常重要。 注 2: 为了更好地测量,在操作场景评审或用户操作测试期间,从易用性角度找出哪个功能受益于监视能力是很有帮助的。这样的功能通常来说也被指定为需求。				
UOp-6-S	撤销操作能力	具有重要结果的任务中可提供重新确认选项或撤销操作的比例是多少?	$X = A / B$ $A =$ 提供撤销操作或重新确认的任务数量 $B =$ 用户能够从重新确认或撤销操作中获益的任务数量	对能提供撤销操作或重新确认的任务数进行计数,并与相关文档(例如需求规格说明、合同等)中规定的期望受益于重新确认或撤销操作的任务数量进行比较

表 14 (续)

ID	名称	描述	测量函数	方法
UOp-7-S	信息分类的易理解性	软件在多大程度上按目标用户所熟悉并方便用户工作的类别管理信息?	$X = A / B$ A = 对于预期用户来说, 熟悉和方便的信息结构数量 B = 使用的信息结构数量	通过对信息分类进行评价, 确定用户所熟悉的信息结构数量, 并与要求的信息结构数量进行比较
示例: 百货商店的网上商店以与在实体店中布局商品的相似方式组织商品。				
UOp-8-S	外观一致性	具有相似项的用户界面中拥有相似外观的比例是多少?	$X = 1 - A / B$ A = 具有相似项但外观不同的用户界面的数量 B = 具有相似项的用户界面的数量	通过对相似项的用户界面数量进行评价, 确定具有相似项但外观不同的用户界面数量, 并与要求的用户界面数量进行比较
注: 例如, 屏幕中的“确定”和“取消”一直显示在同样的位置。				
UOp-9-S	输入设备的支持性	通过所有适当的输入方法(例如键盘、鼠标或语音)启动任务的程度如何?	$X = A / B$ A = 可由所有适当的输入方法启动任务的数量 B = 系统支持的任务数量	对可由键盘、鼠标或语音等输入方法启动任务进行计数, 并与系统支持的任务数进行比较
示例: 在搜索窗体中, 通过使用鼠标或键盘的“Enter”按键激活搜索按钮。				

8.5.5 用户差错防御性测度

用户差错防御性测度用于评估产品或系统预防用户犯错的程度(见表 15)。

注: 用户差错防御性测度是通过对操作员代表或最终用户代表的操作性测试进行测量, 或者通过静态分析例如对需求、设计规格说明或用户手册进行评审来测量。

表 15 用户差错防御性测度

ID	名称	描述	测量函数	方法
UEp-1-G	抵御误操作	有多少比例的用户操作和输入可以防止导致系统故障?	$X = A / B$ A = 实际操作中可以防止导致系统故障的用户操作和输入的数量; B = 可以防止导致系统故障的用户操作和输入的数量	对能避免由于误操作导致关键和严重失效的操作计数, 并与要求的误操作数量进行比较
注 1: 抵御误操作包括在执行无法撤销并且会产生严重后果的操作之前要求系统进行确认。 示例: 删除应用程序中的文件时, 需要用户确认每次删除操作。 注 2: 为了更好地进行测量, 找出用户在操作测试中经常出错的操作和输入是很有帮助的。对于错误的用户操作和输入的预防也可以规定为要求。				

表 15（续）

ID	名称	描述	测量函数	方法
UEp-2-S	用户输入差错纠正率	系统在多大程度上为检测到的用户输入差错提供正确值，并给出原因？	$X = A / B$ A = 系统提供建议的修改值的输入差错数量 B = 检测到的输入差错数量	对系统可提供建议的修改值的输入差错的数量计数，并与检测到的输入差错数量进行比较
注：有关数据质量的详细信息，参考 GB/T 25000.24—2017 中的 Cre-I-1。				
UEp-3-S	用户差错易恢复性	系统可纠正或恢复用户差错的比例是多少？	$X = A / B$ A = 由系统恢复的用户差错数量，这些用户差错是经设计并测试的 B = 操作过程中可能发生的用户差错数量	对系统恢复的用户差错进行计数，并与操作中可能发生的用户差错数量进行比较

8.5.6 用户界面舒适性测度

用户界面舒适性测度用于评估用户界面提供令人愉悦和满意的交互的程度（见表 16）。

表 16 用户界面舒适性测度

ID	名称	描述	测量函数	方法
UIn-1-S	用户界面外观舒适性	用户界面和整体设计在外观舒适上令人愉悦的程度如何？	$X = A / B$ A = 在外观舒适性上令人愉悦的显示界面数量 B = 显示界面数量	通过对显示界面进行评审，确定外观上令人愉悦的显示界面的数量，并与要求的显示界面数量进行比较
注 1：内部或外部用户界面舒适性测度用于评估用户界面的外观，并将受到诸如屏幕设计和颜色等因素的影响。这一点对消费类产品来说是尤为重要。 注 2：良好的配色方案可以帮助用户快速阅读文本或识别图像，因此它有助于用户界面舒适性测量以解决不良的配色方案，例如灰色中的浅蓝色、橙色中的红色、蓝色中的绿色等。 注 3：这个质量测度经常依赖于用户个体，因此专业的易用性设计者或代表用户的测试人员，或来自特定用户群体中的代表，都将参与到该测度的测量。				

8.5.7 易访问性测度

易访问性测度用于评估在指定使用周境中，为了达到指定的目标，产品或系统被具有最广泛的特征和能力的个体所使用的程度（见表 17）。

注：对于易访问性的附加准则，可参考 ISO 9241-171。

表 17 易访问性测度

ID	名称	描述	测量函数	方法
UAc-1-G	特殊群体的易访问性	特殊群体用户成功使用系统的程度如何(如适用,使用辅助技术)?	$X=A/B$ A=特殊群体用户成功使用的功能数量 B=实现的功能数量	选取具有特殊需求的用户群体进行测试,通过执行功能确定被顺利执行和使用功能的数量,并与要求的功能数量进行比较
注 1: 特殊群体包括认知障碍、生理缺陷、听觉/语音障碍和视觉障碍的用户。 注 2: 能力范围包括与年龄相关的障碍。 注 3: 任何人都有可能成为认知、生理、特定情况或环境下的听觉或视觉能力上受限的用户,例如在黑暗中、在高海拔的低气压中、在水中等。				
UAc-2-S	支持的语种充分性	能支持多少种不同的语种?	$X=A/B$ A=实际支持的语种数量 B=需要支持的语种数量	对系统所支持的语种进行计数,并与规定的语种数量进行比较
注: 当用户试图使用与其母语不同的语种的系统或软件时,用户经常会操作错误,并且有时会放弃实现预期目标。这是可访问性降低的一种情况,是由于对描述和消息的误解造成的。因此,有必要考虑和实施支持哪些语种以适应用户的多样性。				

8.5.8 易用性的依从性测度

易用性的依从性测度用于评估产品或系统遵循与易用性相关的标准、约定或法规以及类似规定的程度(见表 18)。

表 18 易用性的依从性测度

ID	名称	描述	测量函数	方法
UCl-1-G	易用性的依从性	遵循与产品或系统易用性有关的标准、约定或法规以及类似规定的程度如何?	$X=A/B$ A=在评价中已证实的正确实现与易用性的依从性相关的项数 B=与易用性的依从性相关的项数	依据与易用性有关的依从性标准、约定或法规以及类似规定,对已满足所要求的依从性(例如在易学性、易操作性方面)的项数进行计数,并与需求规格说明或其他相关文档中要求的易用性的依从性项数相比较

8.6 可靠性测度

8.6.1 可靠性测度概述

可靠性测度用于评估系统、产品或组件在指定条件下、指定时间内执行指定功能的程度。

在系统/软件产品的开发期间,内部可靠性测度用于预测该系统/软件产品是否满足规定的可靠性要求。

外部可靠性测度用于对这样一些属性进行评估,在执行测试过程中,它们与软件作为其一部分的系统的行为有关,以表明在系统运行过程中软件的可靠性程度。在大多数情况下,系统与软件不互相区分。

8.6.2 成熟性测度

成熟性测度用于评估系统、产品或组件在正常运行时满足可靠性要求的程度(见表 19)。

注:成熟性的概念也可应用到其他的质量特性中,以表明它们在正常运行时满足需求的程度(见 GB/T 25000.10—2016)。

表 19 成熟性测度

ID	名称	描述	测量函数	方法
RMa-1-G	故障修复率	检测到与可靠性相关的故障中已被修复的比例是多少?	$X = A / B$ A = 设计/编码/测试阶段修复的与可靠性相关故障数 B = 设计/编码/测试阶段检测到的与可靠性相关的故障数	对在设计/编码/测试阶段已经修复的与可靠性相关的故障数进行计数,并与已检测到的故障总数相比较
注:例如不充分的差错处理是一种与可靠性相关的故障。				
RMa-2-G	平均失效间隔时间(MTBF)	在系统/软件运行过程中平均失效间隔时间是多少?	$X = A / B$ A = 运行时间 B = 实际发生的系统/软件失效次数	对系统/软件在一定的运行时间内发生失效的次数进行计数,并计算失效的平均间隔时间
注 1: 结果值介于 0 到无穷大之间,通常来说结果值越大越好。 注 2: 平均失效间隔时间可以用于比较不同系统或软件产品的可靠性。				
RMa-3-G	周期失效率	在一个预定义的周期内发生失效的数量是多少?	$X = A / B$ A = 在观察时间内检测到的失效数量 B = 观察持续周期数	预定义周期长度,对系统/软件在一定的周期长度内检测到的失效次数进行计数,并计算平均失效数量
注 1: 该测度中使用的周期指实际使用时间或测试时间。依据测试和操作目的,该周期可能不同。 注 2: 可靠性估计模型可使用该测度作为输入。 注 3: 该质量测度的有效性依赖于测试用例的充分性(例如正常测试用例、例外测试用例、异常测试用例)或测试过程中系统使用的范围。				

表 19 (续)

ID	名称	描述	测量函数	方法
RMa-4-S	测试覆盖率	实际执行的系统或软件能力、运行场景或功能与预期的系统或软件能力、运行场景或功能的比例是多少?	$X = A / B$ A = 实际所执行的系统或软件能力、运行场景或功能的数量 B = 预期包含的系统或软件能力、运行场景或功能的数量	对在测试期间系统或软件能力(包括响应时间、容量等)、运行场景或功能进行计数,并与为获得充分的测试覆盖率而要求的系统或软件能力、运行场景或功能数相比较

8.6.3 可用性测度

可用性测度用于评估系统、产品或组件在需要使用时能够进行操作和访问的程度(见表 20)。

表 20 可用性测度

ID	名称	描述	测量函数	方法
RAv-1-G	系统可用性	在计划的系统运行时间中,系统实际可用时间的比例是多少?	$X = A / B$ A = 实际提供的系统运行时间 B = 操作计划中规定的系统运行时间	依据业务场景,在特定时间周期内,测量系统因故障不能用后,每次的修复时间,并将系统实际提供的运行时间与规定的运行时间相比较
注:除了常规运行时间外,这种测度可以延长到特殊时间,例如假期和周末。				
RAv-2-G	平均宕机时间	失效发生时,系统不可用的时间是多少?	$X = A / B$ A = 总的宕机时间 B = 观察到的宕机数量	对在特定的时间周期内检测到的宕机次数进行计数,并测量每次系统不能用的宕机时间,计算其平均时间
注 1: 结果值在 0 到无穷大之间,通常来说结果值越小越好。 注 2: 可用性可以通过系统、产品或组件在可使用状态的时间占据总时间的比例进行外部评估。因此,可用性是成熟性(控制失效频率)、容错性和易恢复性(控制每次失效后宕机时间长度)的组合。				

8.6.4 容错性测度

容错性测度用于评估当存在硬件或软件故障时,系统、产品或组件的运行符合预期的程度(见表 21)。

注: 内部或外部容错性测度与发生运行故障或违反规定接口时,系统/软件产品维持规定性能水平的能力有关。

表 21 容错性测度

ID	名称	描述	测量函数	方法
RFt-1-G	避免失效率	能控制多少种故障模式(以测试用例为单位)以避免关键或严重的失效?	$X=A/B$ A=避免发生关键和严重失效的次数(以测试用例为单位计算的数量) B=测试中执行的故障模式(几乎导致失效)的测试用例数量	对已经避免的故障模式的个数(以测试用例为单位)进行计数,并与考虑到的故障模式数(以测试用例为单位)相比较
RFt-2-S	组件的冗余度	为避免系统失效而安装冗余组件比例是多少?	$X=AB$ A=冗余安装系统组件的数量 B=系统组件数量	对为避免系统失效而安装的冗余组件数量进行计数,并与系统组件数量相比较
注: 在一些安全性要求比较高的系统中,可复用控制系统的某些组件以提高系统可靠性。				
RFt-3-S	平均故障通告时间	系统报告故障的发生的快慢程度如何?	$X=\sum_{i=1}^n(A_i-B_i)/n$ A_i = 系统报告故障 i 的时刻 B_i = 故障 i 被检测到的时刻 n =检测到的故障数	在特定的时间周期内测量系统每次故障报告时间和被检测到的时间的差值,计算其平均时间
注: 结果值在 0 到无穷大之间,通常来说结果值越接近 0 越好。				

8.6.5 易恢复性测度

易恢复性测度用于评估发生中断或失效时,产品或系统能够恢复直接受影响的数据并重建期望的系统状态的程度(见表 22)。

表 22 易恢复性测度

ID	名称	描述	测量函数	方法
RRe-1-G	平均恢复时间	软件/系统从失效中恢复需要多长时间?	$X=\sum_{i=1}^n A_i/n$ A_i = 由于第 i 次失效而重新启动,并恢复宕机的软件/系统所花费的总时间 n =发生失效的次数	在特定的时间周期内测量系统每次宕机所花费的全部恢复时间,并计算其平均时间
注 1: 结果值在 0 到无穷大之间,通常来说结果值越小越好。 注 2: 如果将该质量测度同平均恢复时间的目标阈值(需方与供方在约定中所规定的值)作比较时,该质量测度可用于检查符合性。				
RRe-2-S	数据备份完整性	定期备份数据项的比例是多少?	$X=A/B$ A=实际定期备份数据项的数量 B=需要备份的数据项的数量	对在特定的时间周期内实际定期备份的数据项数量进行计数,并与需要备份的数据项数量相比较

8.6.6 可靠性的依从性测度

可靠性的依从性测度用于评估产品或系统遵循与可靠性有关的标准、约定或法规以及类似规定的程度(见表 23)。

表 23 可靠性的依从性测度

ID	名称	描述	测量函数	方法
RCl-1-G	可靠性的依从性	遵循与产品或系统可靠性有关的标准、约定或法规以及类似规定的程度如何?	$X = A / B$ A = 在评价中证实的已正确实现与可靠性依从相关的项数 B = 与可靠性的依从性相关的项数	依据与可靠性有关的依从性标准、约定或法规以及类似规定,对要求的依从性(例如在成熟性、容错性方面)已经满足的项数进行计数,并与需求规格说明或其他相关文档中要求的可靠性的依从性项数相比较

8.7 信息安全性测度

8.7.1 信息安全性测度概述

信息安全性测度用于评估产品或系统保护信息和数据的程度,以使用户、其他产品或系统具有与其授权类型和授权等级一致的数据访问度。

注 1: 渗透测试可用来模拟攻击,因为这种危及信息安全的攻击在通常测试中一般不会发生。

注 2: 从独立系统直至连接互联网的系統,信息安全保障的需求变化很大。确定所需的信息安全功能并确保其有效性已在相关标准中广泛阐明。本部分的用户需要根据风险等级来确定在每种情况下所使用的信息安全功能。

8.7.2 保密性测度

保密性测度用于评估产品或系统确保数据只有在被授权时才能被访问的程度(见表 24)。

表 24 保密性测度

ID	名称	描述	测量函数	方法
SCo-1-G	访问控制性	保密数据项避免未经授权访问的比例是多少?	$X = 1 - A / B$ A = 未经授权可访问的保密数据项的数量 B = 需要访问控制的保密数据项的数量	对未经授权可访问的保密数据项的数量进行计数,将其与需求规格说明或其他相关文档中规定的需要访问控制的保密数据项的数量相比较

表 24 (续)

ID	名称	描述	测量函数	方法
SCo-2-G	数据加密正确性	按照需求规格说明中的要求,实现数据项加密/解密的正确程度如何?	$X = A / B$ $A =$ 正确加密/解密的数据项数量 $B =$ 需要加密/解密的数据项数量	对系统中正确加密/解密数据项的数量进行计数,将其与需求规格说明中的需要加密/解密数据项数量相比较
注: 相关数据质量的详细介绍,可参考 GB/T 25000.24—2017 中 8.9 的 Cnf-I-1。				
SCo-3-S	加密算法的强度	加密算法经过严格审查的比例是多少?	$X = 1 - A / B$ $A =$ 使用时遭到破坏或存在不可接受风险的加密算法的数量 $B =$ 所使用的加密算法的数量	对使用时遭到破坏或存在不可接受风险的加密算法的个数进行计数,并与所使用的加密算法的数量相比较
注 1: 选择经过严格审查的、当前领域专家认为是可靠的算法,以及经过充分测试的算法实现是非常重要的。与某些加密机制一样,源代码需要能够进行分析,例如美国政府系统需要 FIPS140-2 的认证。 注 2: 其他测量加密算法强度的方法,例如使用道德黑客攻击。				

8.7.3 完整性测度

完整性测度用于评估系统、产品或组件防止未授权访问、篡改计算机程序或数据的程度(见表 25)。

表 25 完整性测度

ID	名称	描述	测量函数	方法
SIn-1-G	数据完整性	防止因未经授权访问而造成的数据破坏或篡改的程度如何?	$X = 1 - A / B$ $A =$ 因未经授权访问而破坏或篡改数据项的数量 $B =$ 需要避免数据破坏或篡改的数据项数量	对因未经授权访问而破坏或篡改数据项的数量进行计数,并与需求规格说明或其他相关文档中规定的需要避免数据破坏或篡改的数据项的数量相比较
SIn-2-G	内部数据抗讹误性	采取数据抗讹误性方法的程度如何?	$X = A / B$ $A =$ 实际用于数据抗讹误性方法的数量 $B =$ 可用及推荐的用于数据抗讹误性方法的数量	对实际用于数据抗讹误性方法的数量进行计数,将其与可用及推荐的用于数据抗讹误性方法的数量相比较
注: 数据抗讹误性内部方法包括经常备份数据、定期将数据和参考数据进行比较、在多个镜像站点中存储数据等。				

表 25 (续)

ID	名称	描述	测量函数	方法
SIn-3-S	缓冲区溢出防止率	为防止缓冲区溢出，在软件模块中，对带有用户输入的内存访问已经进行了边界值检查的比例是多少？	$X = A / B$ A = 在带有用户输入的内存访问中，经过边界值检查的访问数量 B = 软件模块中带有用户输入的内存访问数量	对带有用户输入的内存访问中，经过边界值检查的访问进行计数，并与软件模块中带有用户输入的内存访问数量相比较
注：当写入到缓冲区的数据破坏了与目标缓冲区相邻的内存地址中的数据值时，就会产生缓冲区溢出。当从一个缓冲区复制数据到另一个缓冲区时，如果不首先检查数据是否适合目标缓冲区，就会出现这种情况。				

8.7.4 抗抵赖性测度

抗抵赖性测度用于评估活动或事件发生后可以被证实且不可被否认的程度(见表 26)。

表 26 抗抵赖性测度

ID	名称	描述	测量函数	方法
SNo-1-G	数字签名使用率	使用数字签名，处理需要抗抵赖性事务的比例是多少？	$X = A / B$ A = 实际使用数字签名确保抗抵赖性事务的数量 B = 使用数字签名要求抗抵赖性事务的数量	对实际使用数字签名确保抗抵赖性事务的数量进行计数，并与需要使用数字签名要求抗抵赖性事务的数量相比较
注：认证和安全算法也有助于提高抗抵赖性。				

8.7.5 可核查性测度

可核查性测度用于评估实体的活动可以被唯一地追溯到该实体的程度(见表 27)。

表 27 可核查性测度

ID	名称	描述	测量函数	方法
SAc-1-G	用户审计跟踪的完整性	对用户访问系统或数据的审计跟踪的完整程度如何？	$X = A / B$ A = 所有日志中记录的访问次数 B = 对系统或数据的访问次数	对所有日志中记录的访问次数计数，并与对系统或数据的访问次数进行比较

表 27 (续)

ID	名称	描述	测量函数	方法
SAc-2-S	系统日志保留满足度	系统日志存储在稳定存储器中的时间占要求的存储时间的比例是多少?	$X = A / B$ A = 系统日志实际存储在稳定存储器中的时间 B = 要求系统日志存储在稳定存储器中的时间	对系统日志实际存储在稳定存储器中的时间进行测量,并与要求系统日志存储在稳定存储器中的时间相比较
注 1: 稳定存储器是计算机数据存储技术的分类,它保证了任何给定写操作的原子性,并允许编写对某些硬件故障及电源故障具有鲁棒性的软件。通常,通过 RAID 技术在不同磁盘上镜像数据实现稳定的存储功能。 注 2: 结果值介于 0 到无穷大之间,通常来说结果越大于 1 越好。				

8.7.6 真实性测度

真实性测度用于评估对象或资源的身份标识能够被证实符合其声明的程度(见表 28)。

表 28 真实性测度

ID	名称	描述	测量函数	方法
SAu-1-G	鉴别机制的充分性	系统对主体身份的鉴别程度如何?	$X = A / B$ A = 提供鉴别机制的数量(例如用户 ID/密码或 IC 卡) B = 规定的鉴别机制数量	对系统实际提供的鉴别机制的数量进行计数,并与规定的鉴别机制数量相比较
注: 与信息安全性相关的是鉴别模型的强度,以及具有多层次多因素的鉴别和威胁检测能力。所提供协议的因素数量和真实性程度也用于真实性测度。				
SAu-2-S	鉴别规则的符合性	建立所需的鉴别规则的比例是多少?	$X = A / B$ A = 已实现的鉴别规则的数量 B = 规定的鉴别规则的数量	对已实现的鉴别规则的数量进行计数,并与需求规定的鉴别规则的数量相比较

8.7.7 信息安全性的依从性测度

信息安全性的依从性测度用于评估产品或系统遵循与信息安全性有关的标准、约定或法规以及类似规定的程度(见表 29)。

表 29 信息安全性的依从性测度

ID	名称	描述	测量函数	方法
SCI-1-G	信息安全性的依从性	遵循与产品或系统的信息安全性有关的标准、约定或法规以及类似规定的程度如何？	$X = A / B$ A = 在评价中证实的已正确实现与信息安全性依从相关的项数 B = 与信息安全性的依从性相关的项数	依据与信息安全性有关的依从性标准、约定或法规以及类似规定,对要求的依从性(例如在保密性、可核查性方面)已经满足的项数进行计数,并与需求规格说明或其他相关文档中要求的信息安全性的依从性项数相比较

8.8 维护性测度

8.8.1 维护性测度概述

维护性测度用于评估产品或系统能够被预期的维护人员修改的有效性和效率的程度。

8.8.2 模块化测度

模块化测度用于评估由多个独立组件组成的系统或计算机程序,其中一个组件的变更对其他组件的影响最小的程度(见表 30)。

表 30 模块化测度

ID	名称	描述	测量函数	方法
MMo-1-G	组件间的耦合度	系统或计算机程序中组件间存在的依赖关系的强弱程度如何,以及有多少组件不受其他组件更改的影响？	$X = A / B$ A = 所实现的对其他组件没有产生影响的组件数量 B = 需要独立的组件数量	对其他组件没有产生影响的组件数量进行计数,并与在需求规格说明或其他相关文档中规定的需要独立的组件数量相比较
注：阈值有助于确定由于其他组件变化所产生的影响是否降低到最低。设置阈值的例子包括：因其他组件的更改而为该组件中的更改频率设置阈值,或者为该组件直接访问外部共享数据库的数量设置阈值。				
MMo-2-S	圈复杂度的充分性	具有可接受的圈复杂度的软件模块数量是多少？	$X = 1 - A / B$ A = 圈复杂度的得分超过规定阈值的软件模块数量 B = 已实现的软件模块数量	对圈复杂度的得分超过规定阈值的软件模块数量进行计数,并与已实现的软件模块数量相比较
注：阈值是用于确定每个模块的圈复杂度值是否是可接受,它在每个项目或组织中定义,对于不同编程语言、模块或函数类型,该阈值也可能不同。				

8.8.3 可重用性测度

可重用性测度用于评估资产能够被用于多个系统,或其他资产建设的程度(见表 31)。

表 31 可重用性测度

ID	名称	描述	测量函数	方法
MRe-1-G	资产的可重用性	系统中可重复使用资产的数量是多少?	$X = A / B$ A = 为可重复使用而设计和实现的资产的数量 B = 系统中资产的数量	对为可重复使用而设计和实现的资产数量进行计数,并与系统中资产数量相比较
注: 在这个测度中,资产可以是工作产品例如需求文档、源代码模块、测试模块、特定硬件资源等。				
MRe-2-S	编码规则符合性	符合所要求编码规则的模块的数量是多少?	$X = A / B$ A = 符合特定系统编码规则的软件模块数量 B = 已实现的软件模块数量	对符合特定系统编码规则的软件模块数量进行计数,并与已实现的软件模块数量相比较
注 1: 特定系统的编码规则可能包括有助于例如模块化、可追踪性和简洁性的规则。 注 2: 该质量测度也可适用于不同特性和子特性,例如易分析性。				

8.8.4 易分析性测度

易分析性测度用于评估预期变更(变更产品或系统的一个或多个部分)对产品或系统的影响、诊断产品或系统的缺陷或失效原因、识别待修改部分的有效性和效率的程度(见表 32)。

表 32 易分析性测度

ID	名称	描述	测量函数	方法
MAn-1-G	系统日志完整性	系统将其操作记录在日志中的程度如何,以便它们可以追踪?	$X = A / B$ A = 实际记录在系统中的日志条数 B = 操作期间审计跟踪所需的日志条数	对在特定的时间周期内实际记录在系统中的日志条数进行计数,并与操作期间审计跟踪所需的日志条数(可根据相关文档的要求,确定所需操作的日志条数)相比较
MAn-2-S	诊断功能有效性	满足原因分析需求的诊断功能比例是多少?	$X = A / B$ A = 对原因分析有效的诊断功能数量 B = 已实现的诊断功能数量	对原因分析有效的诊断功能数量进行计数,并与已实现的诊断功能数量相比较
MAn-3-S	诊断功能充分性	所需诊断功能的实现比例是多少?	$X = A / B$ A = 已实现的诊断功能数量 B = 需要实现的诊断功能数量	对已实现的诊断功能数量进行计数,并与需要实现的诊断功能数量相比较
注: 易分析性测度可用来评估维护方或用户试图诊断软件产品的缺陷,或失效的原因,或标识需要修改的部分时,所需的工作量或耗费的资源。				

8.8.5 易修改性测度

易修改性测度用于评估产品或系统可以被有效地、高效地修改,且不会引入缺陷或降低现有产品质量的程度(见表 33)。

注: 易修改性测度宜对这样一组属性进行测量,即当试图实施规定的修改时,通过维护方、用户或包含该软件的系统的行为来测量维护方或用户的工作量。

表 33 易修改性测度

ID	名称	描述	测量函数	方法
MMd-1-G	修改的效率	与预期时间相比,修改的效率如何?	$X = \sum_{i=1}^n (A_i / B_i) / n$ A_i = 对一个指定类型的修改 i 所消耗的总工作时间 B_i = 对一个指定类型的修改 i 所消耗的预期时间 n = 测量的修改数量	在特定的时间周期内测量指定类型修改的实际时间,并与其预期时间相比较
注 1: 测度结果值 X 大于 1 表示修改效率低,小于 1 则表示修改效率高。 注 2: 特定类型修改的预期时间可以基于历史数据或行业内平均值进行确定。				
MMd-2-G	修改的正确性	已正确实施的修改所占比例是多少?	$X = 1 - A / B$ A = 在实施后的规定时间内,导致事故或失效发生的修改数量 B = 实施的修改数量	在实施后的规定时间内对导致事故或失效发生的修改数量进行计数,并与实施的修改数量相比较
MMd-3-S	修改的能力	在指定的持续时间内进行所需修改的程度如何?	$X = A / B$ A = 在指定的持续时间内实际做出修改的项目数 B = 在指定的持续时间内要求修改的项目数	在指定的持续时间内对实际做出修改的项目数的数量进行计数,并与要求修改的项目数的数量相比较

8.8.6 易测试性测度

易测试性测度用于评估能够为系统、产品或组件建立测试准则,并通过测试执行来确定测试准则是否被满足有效性和效率的程度(见表 34)。

注 1: 内部易测试性测度指明了一组属性,这组属性用于预测系统/软件产品中已设计并实现的自动测试辅助功能的总量。

注 2: 外部易测试性测度宜对这样一些属性进行测量,即当试图测试已修改或未修改的软件时,通过维护方、用户或包含软件的系统的行为来测量维护方或用户的工作量。

表 34 易测试性测度

ID	名称	描述	测量函数	方法
MTe-1-G	测试功能的完整性	已实现的测试功能完整性程度如何？	$X = A / B$ A = 按照规定已实现的测试功能数量 B = 需要的测试功能的数量	对按照规定已实现的测试功能数量进行计数,并与需要的测试功能的数量相比较
MTe-2-S	测试独立性	软件测试独立性的程度如何？	$X = A / B$ A = 在依赖其他系统测试时,能被桩模拟的测试数量 B = 依赖其他系统的测试数量	对依赖于其他系统,通过桩模块进行模拟测试的数量进行计数,并与依赖于其他系统的测试数量相比较
注：桩是软件模块的框架或特殊用途的实现,用于开发或测试一个模块。该模块以调用或其他方式依赖于桩。				
MTe-3-S	测试的重启动性	维护后,能否容易地从重启动点运行测试？	$X = A / B$ A = 在逐步检测的期望点,维护方能够暂停并重启执行中的测试运行的事例数 B = 执行中的测试运行能被暂停的事例数	对在逐步检测的期望点,维护方能够暂停并重启执行中的测试运行的事例数进行计数,并与执行中的测试运行能被暂停的事例数相比较

8.8.7 维护性的依从性测度

维护性的依从性测度用于评估产品或系统遵循与维护性有关的标准、约定或法规以及类似规定的程度(见表 35)。

表 35 维护性的依从性测度

ID	名称	描述	测量函数	方法
MCI-1-G	维护性的依从性	遵循与产品或系统的维护性有关的标准、约定或法规以及类似规定的程度如何？	$X = A / B$ A = 在评价中证实的已正确实现与维护性依从相关的项数 B = 与维护性的依从性相关的项数	依据与维护性有关的依从性标准、约定或法规以及类似规定,对要求的依从性(例如在易分析性、易修改性方面)已经满足的项数进行计数,并与需求规格说明或其他相关文档中要求的维护性的依从性项数相比较

8.9 可移植性测度

8.9.1 可移植性测度概述

可移植性测度用于评估系统、产品或组件能够从一种硬件、软件或者其他运行(或使用)环境迁移到另一种环境的有效性和效率的程度。

8.9.2 适应性测度

适应性测度用于评估产品或系统能够有效地、高效地适应不同的或演变的硬件、软件或者其他运营(或使用)环境的程度(见表 36)。

表 36 适应性测度

ID	名称	描述	测量函数	方法
PAd-1-G	硬件环境的适应性	软件或系统是否能够适应不同的硬件环境?	$X = 1 - A/B$ A = 测试期间未完成或结果没有达到要求的功能数量 B = 不同硬件环境中需要测试的功能数量	对测试期间未完成或结果没有达到要求的功能数量进行计数,并与不同硬件环境中需要测试的功能数量相比较
PAd-2-G	系统软件环境的适应性	软件或系统是否能够适应不同的系统软件环境?	$X = 1 - A/B$ A = 测试期间未完成或结果没有达到要求的功能数量 B = 不同系统软件环境下需要测试的功能数量	对测试期间未完成或对测试中未完成或结果没有达到要求的功能数量进行计数,并与不同系统软件环境下需要测试的功能数量相比较
注 1: 为满足特定的适应性需求,用户须应用适应性规程而不是软件之前所提供的功能,此时须测量该过程所需的用户工作量。 注 2: 系统软件可能包括操作系统、中间件、数据库管理系统、编译器、网络管理系统等。				
PAd-3-S	运营环境的适应性	软件或系统是否能够适应不同的运营环境?	$X = 1 - A/B$ A = 在带有用户环境的运营测试中,测试期间没有完成或结果没有达到要求的功能数量 B = 在不同运营环境中测试的功能数量	在带有用户环境的运营测试中(除硬件、软件环境之外的环境,例如组织环境),测试期间对没有完成或结果没有达到要求的功能数量进行计数,并与在不同运营环境中测试的功能数量相比较

8.9.3 易安装性测度

易安装性测度用于评估在指定环境中,产品或系统能够成功地安装和/或卸载的有效性和效率的程度(见表 37)。

表 37 易安装性测度

ID	名称	描述	测量函数	方法
PIn-1-G	安装的时间效率	与预期安装时间相比,实际安装的效率如何?	$X = \sum_{i=1}^n (A_i / B_i) / n$ A_i = 第 i 次安装所消耗的总工作时间 B_i = 第 i 次安装的预期时间 n = 测量的安装次数	测量实际安装所消耗的时间,并与其安装的预期时间相比较
注 1: 结果值 X 大于 1 表示安装效率较低,小于 1 则表示安装效率较高。 注 2: 安装的预期时间可以基于历史数据或行业内平均值进行确定。				
PIn-2-G	安装的灵活性	为使用方便,用户或维护方是否可以自定义安装规程?	$X = A / B$ A = 用户成功自定义安装规程的数量 B = 为使用方便,用户尝试自定义安装规程的数量	对用户成功自定义安装规程的数量进行计数,并与为使用方便,用户尝试自定义安装规程的数量相比较
注: 安装规程的改变可认为是用户的自定义安装。				

8.9.4 易替换性测度

易替换性测度用于评估在相同的环境中,产品能够替换另一个相同用途的指定软件产品的程度(见表 38)。

表 38 易替换性测度

ID	名称	描述	测量函数	方法
PRé-1-G	使用相似性	原软件产品被替换后,本软件产品的用户功能中有多少功能可以在没有额外学习或变通的情况下执行?	$X = A / B$ A = 替换原软件产品后,本软件产品在没有任何额外的学习或变通的情况下,能够执行的用户功能数量 B = 替换原软件产品后,本软件产品中用户功能的数量	替换原软件产品后,对本软件产品在没有任何额外的学习或变通的情况下,能够执行的用户功能数量进行计数,并与本软件产品中用户功能的数量相比较
注: 用户功能是指用户可以调用并用于执行包括用户界面在内的预期任务的功能。				
PRé-2-S	产品质量等价性	原软件产品被替换后,满足要求的质量测度的比例是多少?	$X = A / B$ A = 优于或等于被替换产品的新产品质量测度数量 B = 被替换软件产品中的质量测度数量	对优于或等于被替换产品的新产品质量测度数量进行计数,并与被替换软件产品中的质量测度数量相比较
注: 与易替换性相关的重要产品质量包括互操作性、信息安全性和性能效率。				

表 38 (续)

ID	名称	描述	测量函数	方法
Pre-3-S	功能的包容性	原软件产品被替换后,类似功能能否容易被使用?	$X = A / B$ A = 结果与被替换软件产品相似的产品功能数量 B = 被替换软件产品中需要使用的功能数量	对结果与被替换软件产品相似的产品功能数量进行计数,并与被替换软件产品中需要使用的功能数量相比较
Pre-4-S	数据复用/导入能力	原软件产品被替换后,相同的数据能否继续使用?	$X = A / B$ A = 能像被替换软件产品一样继续使用的数据数量 B = 被替换软件产品中需要继续使用的数据数量	原软件产品被替换后,对能像被替换软件产品一样继续使用的数据数量进行计数,并与被替换软件产品中需要继续使用的数据数量相比较

8.9.5 可移植性的依从性测度

可移植性的依从性测度用于评估产品或系统遵循与可移植性有关的标准、约定或法规以及类似规定的程度(见表 39)。

表 39 可移植性的依从性测度

ID	名称	描述	测量函数	方法
PCp-1-G	可移植性的依从性	遵循与产品或系统的可移植性有关的标准、约定或法规以及类似规定的程度如何?	$X = A / B$ A = 在评价中证实的已正确实现与可移植性依从相关的项数 B = 与可移植性的依从性相关的项数	依据与可移植性有关的依从性标准、约定或法规以及类似规定,对要求的依从性(例如在适应性、易安装性方面)已经满足的项数进行计数,并与需求规格说明或其他相关文档中要求的可移植性的依从性项数相比较

附 录 A
(资料性附录)

本部分与 ISO/IEC 25023:2016 的结构性差异

本部分与 ISO/IEC 25023：2016 相比在结构上有部分调整，具体章条编号对照情况见表 A.1。

表 A.1 本部分与 ISO/IEC 25023:2016 的章条编号对照情况

本部分章条编号	对应 ISO/IEC 25023:2016 章条编号
8.2.1	8.2
8.2.2	8.2.1
8.2.3	8.2.2
8.2.4	8.2.3
8.2.5	—
8.3.1	8.3
8.3.2	8.3.1
8.3.3	8.3.2
8.3.4	8.3.3
8.3.5	—
8.4.1	8.4
8.4.2	8.4.1
8.4.3	8.4.2
8.4.4	—
8.5.1	8.5
8.5.2	8.5.1
8.5.3	8.5.2
8.5.4	8.5.3
8.5.5	8.5.4
8.5.6	8.5.5
8.5.7	8.5.6
8.5.8	—
8.6.1	8.6
8.6.2	8.6.1
8.6.3	8.6.2
8.6.4	8.6.3
8.6.5	8.6.4
8.6.6	—
8.7.1	8.7
8.7.2	8.7.1

表 A.1（续）

本部分章条编号	对应 ISO/IEC 25023:2016 章条编号
8.7.3	8.7.2
8.7.4	8.7.3
8.7.5	8.7.4
8.7.6	8.7.5
8.7.7	—
8.8.1	8.8
8.8.2	8.8.1
8.8.3	8.8.2
8.8.4	8.8.3
8.8.5	8.8.4
8.8.6	8.8.5
8.8.7	—
8.9.1	8.9
8.9.2	8.9.1
8.9.3	8.9.2
8.9.4	8.9.3
8.9.5	—
附录 A	—
附录 B	附录 A
附录 C	附录 B
附录 D	附录 C
D.2.3.1	C.2.3
D.2.3.2	C.2.3.1
D.2.3.3	C.2.3.2
D.2.3.4	C.2.3.3
D.4.1	C.4
D.4.2	C.4.1
D.4.3	C.4.2
D.4.4	C.4.3
D.4.5	C.4.4
D.4.6	C.4.5
D.4.7	C.4.6
D.4.8	C.4.7
D.4.9	C.4.8

表 A.2 给出了本部分与 ISO/IEC 25023：2016 的表格编号对照情况。

表 A.2 本部分与 ISO/IEC 25023：2016 的表格编号对照情况

本部分表格编号	对应 ISO/IEC 25023:2016 表格编号
表 1	表 1
表 2	表 2
表 3	表 3
表 4	—
表 5	表 4
表 6	表 5
表 7	表 6
表 8	—
表 9	表 7
表 10	表 8
表 11	—
表 12	表 9
表 13	表 10
表 14	表 11
表 15	表 12
表 16	表 13
表 17	表 14
表 18	—
表 19	表 15
表 20	表 16
表 21	表 17
表 22	表 18
表 23	—
表 24	表 19
表 25	表 20
表 26	表 21
表 27	表 22
表 28	表 23
表 29	—
表 30	表 24
表 31	表 25
表 32	表 26
表 33	表 27
表 34	表 28
表 35	—

表 A.2（续）

本部分表格编号	对应 ISO/IEC 25023:2016 表格编号
表 36	表 29
表 37	表 30
表 38	表 31
表 39	—
表 A.1	—
表 A.2	—
表 B.1	表 A.1

附录 B
(资料性附录)
质量测度使用时的考虑

本附录给出了质量测度选择和应用的说明,定义在第 8 章中的每一个质量测度都可以用于测量内部属性(典型的是对中间产品的静态测度)、外部属性(典型的是通过测量代码执行时的行为),或者内外部属性。

- 注 1: 当迭代或增量模型应用于开发或维护时,内部和外部测度都可用于迭代或增量的每个周期。迭代地增加或改进系统/软件规格说明、架构设计、详细设计、组件和代码能够通过内部测度的评审方法进行测量。然而迭代集成的系统/软件能够通过每次迭代或增量期间执行外部测度构建的测试任务进行测量。如果可执行测试能够在迭代或增量期间频繁地进行,则可能选择外部测度(或者使用质量测度)而不是内部测度。然后,可以通过多次迭代或增量重复使用这些质量测度来监控不断变化的质量趋势。例如,功能覆盖率的测量值(功能性的质量测度之一)在早期迭代中可能较低,预计在后续迭代中该值会逐渐增加。
- 注 2: 性能效率的内部测度适用于静态设计文档或源代码。这些测量值可通过预估设计算法的理论计算量、函数调用的次数或可执行代码的步骤来获得。然而,在设计过程中,对中间可执行原型的性能效率的外部测度的应用有助于理解内部测度和外部测度的实际差距,并校准内部测度的估计值。
- 注 3: 易用性的内部测度适用于屏幕显示的静态模型、易用性设计规格说明、一组消息文本文件、用户手册、用户界面源代码等。然而,在开发过程中,对中间可执行原型的易用性的外部测度的应用有助于理解内部测度和外部测度的实际差距。如果可用,在开发过程中,将使用质量测度应用于可执行原型也是很有帮助的。

此外,质量测度可以根据推荐等级进行分类:

- HR:高度推荐,表示经常使用这个质量测度;
- R:推荐,表示在合适的时候使用这个质量测度;
- UD:用户自行判断使用,表示开发一个新的质量测度时将这个质量测度作为参考,因为新的质量测度具有未知的可靠性。

表 B.1 给出了使用每个质量测度的相关说明。

表 B.1 质量测度使用总结

质量特性	质量子特性	ID	质量测度名称	内/外	推荐等级
功能性	功能完备性	FCp-1-G	功能覆盖率	内外	HR
	功能正确性	FCr-1-G	功能正确性	内外	HR
	功能适合性	FAp-1-G	使用目标的功能适合性	内外	HR
		FAp-2-G	系统的功能适合性	内外	HR
	功能性的依从性	FCI-1-G	功能性的依从性	内外	HR
性能效率	时间特性	PTb-1-G	平均响应时间	内外	HR
		PTb-2-G	响应时间的充分性	内外	R
		PTb-3-G	平均周转时间	内外	R
		PTb-4-G	周转时间充分性	内外	R
		PTb-5-G	平均吞吐量	内外	R

表 B.1 (续)

质量特性	质量子特性	ID	质量测度名称	内/外	推荐等级
性能效率	资源利用性	PRu-1-G	处理器平均占用率	外	HR
		PRu-2-G	内存平均占用率	外	R
		PRu-3-G	I/O 设备平均占用率	外	R
		PRu-4-G	带宽占用率	外	UD
	容量	PCa-1-G	事务处理容量	内外	R
		PCa-2-G	用户访问量	内外	R
		PCa-3-G	用户访问增长的充分性	外	UD
	性能效率的依从性	PCl-1-G	性能效率的依从性	内外	HR
兼容性	共存性	CCo-1-G	与其他产品的共存性	外	HR
	互操作性	CIn-1-G	数据格式可交换性	内外	HR
		CIn-2-G	数据交换协议充分性	内外	R
		CIn-3-S	外部接口充分性	内外	HR
	兼容性的依从性	CCl-1-G	兼容性的依从性	内外	HR
易用性	可辨识性	UAp-1-G	描述的完整性	内外	HR
		UAp-2-S	演示覆盖率	内外	UD
		UAp-3-S	入口点的自描述性	内外	UD
	易学性	ULe-1-G	用户指导完整性	内外	HR
		ULe-2-S	输入字段的默认值	内外	R
		ULe-3-S	差错信息的易理解性	内外	R
		ULe-4-S	用户界面的自解释性	内外	UD
	易操作性	UOp-1-G	操作一致性	内外	HR
		UOp-2-G	消息的明确性	内外	R
		UOp-3-S	功能的易定制性	内外	UD
		UOp-4-S	用户界面的易定制性	内外	UD
		UOp-5-S	监视能力	内外	UD
		UOp-6-S	撤销操作能力	内外	R
		UOp-7-S	信息分类的易理解性	内外	R
		UOp-8-S	外观一致性	内外	UD
		UOp-9-S	输入设备的支持性	内外	UD
	用户差错防御性	UEp-1-G	抵御误操作	内外	HR
		UEp-2-S	用户输入差错纠正率	内外	HR
		UEp-3-S	用户差错易恢复性	内外	R
	用户界面舒适性	UIn-1-S	用户界面外观舒适性	内外	UD

表 B.1 (续)

质量特性	质量子特性	ID	质量测度名称	内/外	推荐等级
易用性	易访问性	UAc-1-G	特殊群体的易访问性	内外	R
		UAc-2-S	支持的语种充分性	内外	UD
	易用性的依从性	UCI-1-G	易用性的依从性	内外	HR
可靠性	成熟性	RMa-1-G	故障修复率	内外	HR
		RMa-2-G	平均失效间隔时间 (MTBF)	外	HR
		RMa-3-G	周期失效率	外	R
		RMa-4-S	测试覆盖率	外	R
	可用性	RAv-1-G	系统可用性	外	HR
		RAv-2-G	平均宕机时间	外	R
	容错性	RFt-1-G	避免失效率	外	HR
		RFt-2-S	组件的冗余度	内外	R
		RFt-3-S	平均故障通告时间	外	UD
	易恢复性	RRe-1-G	平均恢复时间	外	HR
		RRe-2-S	数据备份完整性	内外	R
	可靠性的依从性	RCl-1-G	可靠性的依从性	内外	HR
信息安全性	保密性	SCo-1-G	访问控制性	内外	HR
		SCo-2-G	数据加密正确性	内外	R
		SCo-3-S	加密算法的强度	内外	UD
	完整性	SIn-1-G	数据完整性	内外	HR
		SIn-2-G	内部数据抗讹误性	内外	R
		SIn-3-S	缓冲区溢出防止率	内	UD
	抗抵赖性	SNo-1-G	数字签名使用率	内外	R
	可核查性	SAc-1-G	用户审计跟踪的完整性	内外	HR
		SAc-2-S	系统日志保留满足度	内外	R
	真实性	SAu-1-G	鉴别机制的充分性	内外	HR
		SAu-2-S	鉴别规则的符合性	内外	R
	信息安全的依从性	SCl-1-G	信息安全的依从性	内外	HR
维护性	模块化	MMo-1-G	组件间的耦合度	内外	R
		MMo-2-S	圈复杂度的充分性	内	UD
	可重用性	MRe-1-G	资产的可重用性	内外	HR
		MRe-2-S	编码规则符合性	内	R

表 B.1（续）

质量特性	质量子特性	ID	质量测度名称	内/外	推荐等级
维护性	易分析性	MAn-1-G	系统日志完整性	内外	HR
		MAn-2-S	诊断功能有效性	内外	R
		MAn-3-S	诊断功能充分性	内外	R
	易修改性	MMd-1-G	修改的效率	内外	HR
		MMd-2-G	修改的正确性	内外	HR
		MMd-3-S	修改的能力	内外	UD
	易测试性	MTe-1-G	测试功能的完整性	内外	R
		MTe-2-S	测试独立性	内外	UD
		MTe-3-S	测试的重启动性	内外	UD
	维护性的依从性	MCI-1-S	维护性的依从性	内外	HR
可移植性	适应性	PAd-1-G	硬件环境的适应性	外	HR
		PAd-2-G	系统软件环境的适应性	外	HR
		PAd-3-S	运营环境的适应性	外	UD
	易安装性	PIn-1-G	安装的时间效率	外	R
		PIn-2-G	安装的灵活性	外	R
	易替换性	PRe-1-G	使用相似性	内外	HR
		PRe-2-S	产品质量等价性	内外	R
		PRe-3-S	功能的包容性	外	R
		PRe-4-S	数据复用/导入能力	外	UD
	可移植性的依从性	PCp-1-G	可移植性的依从性	内外	HR

附录 C

(资料性附录)

利用质量测度元素定义产品或系统质量测度

大部分用于各类质量测度的测量函数中的质量测度元素已在 GB/T 25000.21 中附录 A 中给出。如果需要,可以根据 GB/T 25000.21 提供的步骤和表格格式适当地定义或设计一些新的质量测度元素。

下面列出了在不同质量测度的测量函数中经常使用的质量测度元素。

注:更多关于质量测度元素 QME 的信息,见 GB/T 25000.21 中有关 QME 的定义。

C.1 功能数

满足特定 QME 定义中给出条件的所有功能的计数。

注:这些功能可以是,例如必需的、实现的、测试的、必要的、可选的,或它们中的任意组合等。

C.2 失效数

在给定时间内发生的所有失效的计数,这些失效也满足特定 QME 定义中给出的条件。

示例:质量测度元素的例子有预期失效数、检测到的失效数、已解决的失效数、给定严重程度等级的失效数。

C.3 故障数

在给定软件产品组件中检测(或估计)的软件产品故障的计数,这些故障也满足特定 QME 定义中给出的条件,例如给定类别的故障数、给定严重程度等级的故障数、成功纠正的故障数等。

C.4 产品规模

根据期望的准则对软件产品组件的计数。产品规模可以是代码行数(LOC)、功能点、模块、类,或可视化结构(例如图表或其部分)。

注:软件产品组件只有在满足某些附加属性时才能被计数,例如仅可执行代码行、包含注释的代码行、仅注释行、声明或类型转换、仅包含括号/大括号等。

C.5 持续时间

在特定 QME 定义中描述的任何进程的开始时间和结束时间之间的间隔(持续时间 = 结束时间 - 开始时间)。例如,

- 执行时间:由计算机时钟内部测量的时间,例如 CPU 时间、I/O 时间等,或者通过插入代码或软件工具(例如测试套件)测量的时间;
- 观察时间:由观察者在外部使用外部时钟测量的时间,例如完成一个事务或用户任务的时间;
- 设置时间:固定的时间进程或观察,但对与动作无关的测度来说是有意义的,例如所需的响应时间。

C.6 测试用例数

满足特定 QME 定义中给出条件的不同测试输入数据和场景的计数,例如测试用例设计、要求、执行(成功或失败)等。

C.7 重启次数

统计系统在严重失效后恢复计算的尝试次数,并且满足特定 QME 定义中给出的条件。它可以区分系统的重新启动和恢复。

C.8 I/O 数量

统计满足特定 QME 定义中给出条件的 I/O 事件数的计数。对于观察者来说,I/O 事件与系统消息是不同的。

- 观察者与系统之间交互,例如一次对话;
- 事务:观察者和系统之间的交互序列,需要以原子方式执行完成操作,例如向导(带有可选项)。

C.9 任务数

任务是实现给定目标所需的一组或一系列活动。任务数是满足特定 QME 定义中给出条件的任务的计数。它可以区分:

- 用户任务:用户(使用软件产品)针对指定目标执行的活动;
- 系统任务:系统为支持用户而执行的活动。

C.10 用户尝试(试验)次数

执行相同操作以满足特定 QME 定义中给出条件的尝试的计数。这些尝试可以是:

- 评价:具有相同输入和相同场景的迭代(例如压力测试);
- 用例:具有不同输入和/或不同场景的迭代。

C.11 数据项数

满足特定 QME 定义中给出的条件的不同结构、类或数据格式的计数。

C.12 记录数

满足特定 QME 定义中给出的条件的相同结构、类或格式的记录计数。

C.13 需求数

满足特定 QME 定义中给出的条件的需求条款的计数。

注:需求可以是必要的、可选的、确认的,或这些需求的任意组合等。

C.14 用户操作数

由用户执行的且满足特定 QME 定义中给出的条件的操作数量的计数,其中操作是执行任务所需的一系列步骤。

C.15 系统操作数

由系统执行的且满足特定 QME 定义中给出的条件的完整操作的计数。

注:该 QME 计算的是完整操作的数量,而不是每个操作所需的各个步骤。

C.16 语种数

用于执行预期用户功能的系统或软件产品支持的不同语种的计数。

C.17 软件模块数

独立于其他组件工作的软件组件的计数。从概念上讲,模块代表关系的分离,并通过加强组件之间的逻辑边界提高维护性。

C.18 接口数

计算机系统的两个独立组件交换信息的共享边界的计数。交换可以在软件、硬件、外围设备、人和这些的组合之间进行。

附 录 D
(资料性附录)
测量类型的详细说明

D.1 概述

为了设计一个收集数据、正确解释其含义并且把测度规范化以便进行比较的过程,测度的用户宜标识并考虑测量所使用的测度类型。

注:对大多数外部测度来说,通过执行测试可以收集测量函数的输入数据。在此附录中详细解释的测量类型与测试设计技术和 ISO/IEC/IEEE 29119-4 中定义的测试类型密切相关。有关质量相关的测试类型和将质量特性/子特性映射到的测试类型均在 ISO/IEC/IEEE 29119-4:2015 的附录 A 中给出详细描述。

D.2 测度规模类型

D.2.1 导引

按其定义中所声称的测度内容,本类型的测度代表软件的一种特殊规模。

注:软件可以有多种表示规模的方法(就像任何一个实体可以进行多维测量——质量、体积、表面积等)。

用一种规模测度来使其他的测度规范化,可以根据规模单位给出可比值。下列描述的规模测度类型可用于软件质量的测量。

D.2.2 功能规模类型

功能规模是软件可能有的一种规模类型(一维)的例子。任何一个软件实例可能会有多个功能规模,例如取决于:

- 测量软件规模的目的(它影响到在测量中包含的软件范围);
- 所用的特定功能规模测量方法(它将改变其单位和标度)。

ISO/IEC 14143-1 提供了概念定义和使用功能规模的测量方法(FSM 方法)的过程。

为了规范化地使用功能规模,需要确保采用相同的功能规模方法,基于同样的目的,还要确保要比较的不同软件已经过测量,因而具有可比较的范围。

尽管下列内容经常声称代表了功能规模,但不能保证它们等同于应用 FSM 方法所获得的功能规模,也不能保证它们依从于 ISO/IEC 14143-1。不过,在软件开发中,如下的方法仍被广泛使用。

- 电子表格数;
- 屏幕数;
- 要处理的文件或数据集合数;
- 用户需求规格说明描述的逐条列举的功能需求数。

D.2.3 程序规模类型

D.2.3.1 导引

本条中,术语“程序设计”代表当执行时导致一些动作的表达式,术语“语言”代表所用的表达式类型。

D.2.3.2 源程序规模

应解释程序设计语言,它应提供如何处理诸如注释行这样的不可执行语句。经常使用下列测度:
非注释性源语句(NCSS)包括可执行语句和带有逻辑性源语句的数据声明语句。

注 1: 新程序规模:开发方可能使用新开发的程序规模来代表开发与维护工作产品的规模。

注 2: 变更的程序规模:开发方可能使用变更的程序规模来代表包含修改过的组件的软件规模。

可能有必要更详细地区分下列源代码语句的类型:

——语句的类型

- 逻辑性源语句(LSS):LSS 测量软件指令的数量。这些语句不考虑与行的关系,独立于表现它们的物理格式;
- 物理源语句(PSS):PSS 测量软件的源代码行数。

——语句的属性

- 可执行语句;
- 数据声明语句;
- 编译程序命令语句;
- 注释性源语句。

——源语句

- 修改的源语句;
- 增加的源语句;
- 删除的源语句;
- 新开发的源语句(= 增加的源语句+修改的源语句);
- 重用的源语句(= 原来的源语句-修改的源语句-删除的源语句)。

D.2.3.3 程序字计数规模

可采用下列 Halstead 测度方法计算测量值:

程序的词汇数= $n_1 + n_2$;观察到的程序长度= $N_1 + N_2$,其中:

- n_1 :不同操作符的数量(即程序源代码中被程序语言预留的不同操作符的字数);
- n_2 :不同操作数的数量(即程序源代码中由编程人员定义的不同操作数的字数);
- N_1 :操作符的总数量(即程序源代码中不同操作符出现的次数);
- N_2 :操作数的总数量(即程序源代码中不同操作数出现的次数)。

D.2.3.4 模块数

本测量是计算可独立执行的对象个数,例如程序中的模块个数。

D.2.4 利用的资源测度类型

本测度类型标识要评价的软件在运行中所用的资源。例如:

- a) 存储器的数量,例如,在软件执行过程中,临时和永久占用的磁盘或内存的数量;
- b) I/O 负载,例如,通信数据流量总数(对网络中的备份工具有意义);
- c) CPU 负载,例如,每秒钟 CPU 指令集占用的百分比(本测度类型对测量 CPU 的利用率和并发/并行系统中软件的多线程运行时测量进程分配的效率时有意义);
- d) 文件与数据记录,例如,文件或记录的位长度;
- e) 文档,例如,文档页数。

注意峰值(最大值),最小值和平均值,以及时间周期及观察的次数等数据可能很重要。

D.2.5 特定的操作规程步骤类型

本测度类型标识在人工界面的设计规格说明或用户手册中规定的规程的静态步骤。

本测量值可能依测量所用的描述类型的不同而有所区别,例如用户的操作规程可以用图形也可以用文字来表示。

D.3 时间测度类型

D.3.1 导引

时间测度类型测度的用户应记录时间周期、检查过多少站点及有多少用户参与了这一测量。

有多种以时间为单位进行测量的方式,例如:

a) 实时单位

这是物理时间单位,如秒、分或小时。这种单位常用来描述实时软件的任务处理时间。

b) 计算机机器时间单位

这是计算机处理器的时钟时间,即 CPU 时间的秒、分或小时。

c) 正式的日程表上的时间单位

包括工作小时、日历(日、月或年)。

d) 组件的时间单位

在有多个站点时,组件时间单位标识各个站点,组件时间单位是每个站点单独时间的累计。这种单位通常用来描述组件的可靠性,如组件的失效率。

e) 系统时间单位

在有多个站点时,系统时间不标识单独的站点,而标识整个系统中所有运行的站点。这种单位常用来描述系统的可靠性,如系统的失效率。

D.3.2 系统运行时间类型

系统运行时间类型为测量软件的可用性提供了基础。主要用于评价可靠性。应确定软件是间断运行还是连续地运行。如果软件是间断运行的,应确保在软件运行期间对时间进行测量(这显然可以扩展到连续运行的情况)。

a) 经时时间

当软件在不变的情况下使用时,如系统每周运行时间长度相同。

b) 机器加电时间

用于实时的、嵌入的或操作系统软件,它在系统运行的全部时间内都得到充分使用。

c) 规格化的机器时间

类似于机器加电时间,但把多台机器上不同的加电时间数据汇集起来并用一个修正因子进行调整。

D.3.3 执行时间类型

执行时间类型是指为完成特定任务所需要执行软件的时间。应分析几种尝试的分布,应计算均值、方差和最大值。应检查在特定条件下,特别是在过载条件下的执行时间。执行时间类型主要用于评价效率。

D.3.4 用户时间类型

用户时间类型测量单个用户在使用软件完成任务时所花费的时间。例如:

- a) 会话时间
会话开始和结束的时间。如一个家庭银行系统的用户提取钱的行为。对于交互程序来说,只研究交互的易用性问题,不研究空闲时期。
- b) 任务时间
单个用户每次试图运行软件完成任务所花费的时间。应定义好测量的起点和终点。
- c) 用户时间
从开始到某个时间点,用户使用软件所花费的时间(从开始时起,用户使用软件大约有多少小时的时间或天数)。

D.3.5 工作量类型

工作量类型是指与某特定项目任务有关的生产时间。

- a) 个人工作量
开发方、维护方或操作方为完成特定任务进行工作所需要的生产时间。个人的工作量只是每天一定数量的生产小时数。
- b) 任务工作量
任务工作量是指所有单个的项目人员(开发方、维护方、操作方、用户或其他)为完成特定任务进行工作的人员工作量的累计值。

D.3.6 事件的时间间隔类型

本测度类型是指在观察期间,一个事件与下一个事件之间的时间间隔。可用观察时段的频率代替本测度。本测度可以典型地用来描述相继发生的失效之间的时间。

D.4 计数测度类型

D.4.1 导引

若对软件产品的文档属性进行计数,则为静态计数类型。若对事件或人的动作进行计数,则为动态计数类型。

D.4.2 检测的故障数类型

本测量对在评审、测试、纠正、运行或维护期间检测到的故障个数进行计数。按照故障所造成的影响,可为这些故障的严重程度进行分类。

D.4.3 程序结构的复杂度类型

本测量对程序结构的复杂度进行计数。例如不同路径的数目或 McCabe 圈复杂度。

D.4.4 检测不一致的个数类型

本测度对调查所准备的不一致项数进行计数。

- a) 不符合的项数
例如:
 - 与需求规格说明的规定项不相符;
 - 与法律、法规或标准不相符;
 - 与协议、数据格式、介质格式、字符编码不相符;
- b) 用户期望的不能实现实例数

本测量对所列举的满意/不满意的项数进行计数,这些项描述用户合理的期望与软件产品性能间的差别。

本测量可用问卷的方式向测试方、客户、操作方或最终用户就发现的缺陷进行调查。例如:

- 功能是否可用;
- 功能是否有效地执行;
- 功能是否可用于用户特定的预期使用;
- 功能是否是预期的、需要的或不需要的。

D.4.5 变更数类型

本类型标识检测出的已经变更的软件配置项。如在源代码中发生变更的行数。

D.4.6 检测到失效数类型

本测量对在产品开发、测试、运作或维护过程中检测出的失效个数进行计数。根据这些失效造成的影响,可以按严重性的等级进行分类。

D.4.7 尝试(试验)次数类型

本测度对与故障造成的缺陷相关的尝试次数进行计数。例如在评审、测试和维护中的尝试次数。

D.4.8 人工操作过程中的点击类型

当用户与软件在操作中发生互动时,本测度对用户作为的动态步骤活动所产生的点击个数进行计数。本测度量化了人类工效的易用性及使用的工作量。因此,本测度可用于易用性测量。如执行任务时的点击次数,眼睛活动的次数等。

D.4.9 记分类型

本类型标识算术计算的记分或结果。记分可包括计数或按检查表进行或不进行加权计算。例如检查表的记分;问卷调查的记分;Delphi 方法等。

参 考 文 献

- [1] GB/T 25000.12—2017 软件工程 软件产品质量要求和评价(SQaRE) 第 12 部分:数据质量模型
- [2] GB/T 25000.22—2019 系统与软件工程 系统与软件质量要求和评价(SQaRE) 第 22 部分:使用质量测量
- [3] GB/T 25000.24—2017 系统与软件工程 系统与软件质量要求和评价(SQaRE) 第 24 部分:数据质量测量
- [4] GB/T 29831.1~GB/T 29831.3 系统与软件功能性
- [5] GB/T 29832.1~GB/T 29832.3 系统与软件可靠性
- [6] GB/T 29833.1~GB/T 29833.3 系统与软件可移植性
- [7] GB/T 29834.1~GB/T 29834.3 系统与软件维护性
- [8] GB/T 29835.1~GB/T 29835.3 系统与软件效率
- [9] GB/T 29836.1~GB/T 29836.3 系统与软件易用性
- [10] ISO/IEC 9241-11:1998 Ergonomics requirement for office work with visual display terminals(VDTs)—Part 11:Guidance on usability
- [11] ISO/IEC 9241-110 Ergonomics of human-system interaction—Part 110:Dialogue principles
- [12] ISO/IEC 9241-171 Ergonomics of human-system interaction—Part 171:Guidance on software accessibility
- [13] ISO/IEC 14143 Information technology—Software measurement—Functional size measurement
- [14] ISO/IEC 14143-1 Information technology—Software measurement—Functional size measurement—Part 1:Definition of concepts
- [15] ISO/IEC 14756 Information technology—Measurement and rating of performance of computer based software systems
- [16] ISO/IEC 15939:2007 Systems and Software engineering—Measurement process
- [17] ISO/IEC 25020 Software Engineering—Software product Quality Requirements and Evaluation(SQaRE)—Measurement reference model and guide
- [18] ISO/IEC DIS 25030 Software engineering—Software product Quality Requirements and Evaluation (SQaRE)—Quality requirements
- [19] ISO/IEC/IEEE 15288 Systems and software engineering—System life cycle processes
- [20] ISO/IEC/IEEE 24765:2010 Systems and software engineering—Vocabulary
- [21] ISO/IEC/IEEE 29119-4:2015 Systems and software engineering—Software testing—Part 4:Test techniques
- [22] Ministry of Economy, Trade and Industry, Japan (METI). Investigative Report on Measure for System/Software Product Quality Requirement Definition and Evaluation,2011,http://www.meti.go.jp/policy/mono_info_service/joho/cloud/2011/11_05.pdf
- [23] U.S.Department of Health and Human Service.The Research-Based Web Design & Usability Guidelines, Enlarged/Expanded edition.U.S.Government Printing Office, Washington, 2006.
- [24] OMG.CISQ Specification for Automated Quality Characteristic Measures, CISQ-TR-2012-01, 2012.

中 华 人 民 共 和 国
国 家 标 准
系统与软件工程
系统与软件质量要求和评价(SQuaRE)
第 23 部分:系统与软件产品质量测量
GB/T 25000.23—2019

*

中国标准出版社出版发行
北京市朝阳区和平里西街甲 2 号(100029)
北京市西城区三里河北街 16 号(100045)

网址:www.spc.org.cn

服务热线:400-168-0010

2019 年 8 月第一版

*

书号:155066·1-63041

版权专有 侵权必究



GB/T 25000.23-2019