



# 中华人民共和国国家标准

GB/T 16260.2—2006/ISO/IEC TR 9126-2:2003

---

## 软件工程 产品质量 第2部分:外部度量

Software engineering—

Product quality—

Part 2: External metrics

(ISO/IEC TR 9126-2:2003, IDT)

2006-03-14 发布

2006-07-01 实施

中华人民共和国国家质量监督检验检疫总局 发布  
中国国家标准化管理委员会

免费标准网(www.freebz.net) 无需注册 即可下载

## 目 次

前言 .....	I
引言 .....	II
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 符号和缩略语 .....	2
6 软件质量度量的使用 .....	2
7 度量表的阅读和使用 .....	3
8 度量表 .....	4
8.1 功能性度量 .....	4
8.2 可靠性度量 .....	5
8.3 易用性度量 .....	5
8.4 效率度量 .....	6
8.5 维护性度量 .....	7
8.6 可移植性度量 .....	7
附录 A(资料性附录) 使用度量时的考虑 .....	54
附录 B(资料性附录) 使用质量的度量、外部度量和内部度量的用法(框架实例) .....	58
附录 C(资料性附录) 度量标度类型和测度类型的详细解释 .....	64
附录 D(资料性附录) 术语 .....	69

## 前 言

GB/T 16260《软件工程 产品质量》分为如下几部分：

- 第 1 部分(即 GB/T 16260.1):质量模型;
- 第 2 部分(即 GB/T 16260.2):外部度量;
- 第 3 部分(即 GB/T 16260.3):内部度量;
- 第 4 部分(即 GB/T 16260.4):使用质量的度量。

本部分为 GB/T 16260 的第 2 部分。

本部分等同采用 ISO/IEC TR 9126-2:2003《软件工程 产品质量 第 2 部分:外部度量》。

为便于使用,本部分做了下列编辑性修改:

- a) “ISO/IEC 9126 的本部分”改为“本部分”;
- b) 删除了国际标准的前言,修改了国际标准的引言;
- c) 对国际标准 ISO/IEC TR 9126-2:2003 的第 8 章中的部分内容作了编辑性修改,对表述不明确的地方给出了明确表述,即:将第 8 章各度量表中的“5.3 软件集成”明确为“5.3.8 软件集成”、将“5.3 合格性测试”明确为“5.3.9 合格性测试”,以及将“5.4 运行测试”明确为“5.4.2 运行测试”,使之与 GB/T 8566 标准的条目编号一致。去掉了第 8 章中的注 2,保留了注 1。

本部分的附录 A、附录 B、附录 C 和附录 D 是资料性附录。

本部分由中华人民共和国信息产业部提出。

本部分由中国电子技术标准化研究所归口。

本部分起草单位:中国电子技术标准化研究所、上海计算机软件技术开发中心、杭州波导软件有限公司。

本部分主要起草人:王凌、陈森芬、冯惠、韩红强、金荣得、王欣、赵建东。

## 引 言

本部分提供了外部度量,以测量 GB/T 16260.1—2006 中定义的 6 个外部质量特性的属性。本部分所列的度量并非一个完备集。开发者、评价者、质量管理者和需方可以从本部分中选择合适的度量,用来定义质量需求、评价软件产品、测量质量情况或作其他用途;亦可以修改度量或使用本部分未包括的其他度量。本部分适用于各种软件产品,但并非每种度量适用于各种软件产品。

GB/T 16260.1—2006 定义了软件质量特性的术语,及这些特性如何被分解为子特性。但在该部分中并没有描述这些子特性是如何被测量的。对于这些特性或子特性的测量,GB/T 16260.2—2006 定义了外部度量,GB/T 16260.3—2006 定义了内部度量,GB/T 16260.4—2006 定义了使用质量的度量。内部度量用来测量软件本身,外部度量用来测量包含该软件的基于计算机系统的行为,使用质量的度量则测量软件在某个特定使用周境中的使用效果。

本部分旨在与 GB/T 16260.1—2006 一起使用。极力推荐读者在使用本部分之前阅读 GB/T 16260.1—2006 和 GB/T 18905.1—2002,尤其是在读者不熟悉使用软件度量进行产品规格说明和产品评价的情况下。

第 1 章至第 7 章以及附录 A 到附录 D 为 GB/T 16260 的第 2 部分、第 3 部分和第 4 部分共有。

## 软件工程 产品质量

### 第2部分:外部度量

#### 1 范围

GB/T 16260 的本部分定义了依据 GB/T 16260.1—2006 标准定义的特性和子特性来定量测量软件外部质量的外部度量。本部分旨在与 GB/T 16260.1—2006 一起使用。

本部分包括以下内容:

- a) 如何使用软件质量度量的解释;
- b) 每个子特性的基本度量集;
- c) 在软件产品生存周期内如何使用这些度量的实例。

本部分没有为某个评定级别或依从性等级而设置这些度量值的范围,因为这些值是依据每个软件产品或软件产品的一部分的自身特性而定的,也即依赖于软件分类、完整性级别和用户需求等因素。一些属性可能会有期望的取值范围,但不依赖于特定用户的需求,而范围的确定往往依赖于一般因素,例如人类认知因素。

本部分可用于各种应用软件。用户可以选择、修改及应用本部分中的度量和测度,也可以针对独特的应用领域定义特定应用的度量。例如,对于安全性和安全保密性等质量特性的具体测量可参见有关国家标准和国际标准。

本部分旨在针对以下使用者:

- a) 需方(从供方获得或采购系统、软件产品或软件服务的个体或组织);
- b) 评价者(实施评价的个体或组织。例如评价者可以是测试实验室、软件开发组织的质量部门、政府组织或用户);
- c) 开发者(执行开发活动的个体或组织,开发活动包括软件生存周期过程中的需求分析、设计、测试直至验收等活动);
- d) 维护者(执行维护活动的个体或组织);
- e) 供方(按所签合同向需方提供系统、软件产品或软件服务的个体或组织),其在合格性测试中确认软件质量时使用;
- f) 用户(使用软件产品执行具体功能的个体或组织),其在验收测试中评价软件产品质量时使用;
- g) 质量管理者(执行软件产品或软件服务的系统性检查的个体或组织),作为质量保证和质量控制的组成部分评价软件质量时使用。

#### 2 符合性

符合性不作要求。

注:在 GB/T 16260.1—2006 质量模型中有关于度量的一般符合性要求。

#### 3 规范性引用文件

下列文件中的条款通过 GB/T 16260 的本部分的引用而成为本部分的条款。凡是注明日期的引用文件,其随后所有的修改单(不包括勘误的内容)或修订版均不适用于本部分,然而,鼓励根据本部分达成协议各方研究是否可使用这些文件的最新版本。凡是不注明日期的引用文件,其最新版本适用于本部分。

## GB/T 16260.2—2006/ISO/IEC TR 9126-2:2003

GB/T 5271.20—1994 信息技术 词汇 第20部分:系统开发 (eqv ISO/IEC 2382-20:1990)

GB/T 8566—2001 信息技术 软件生存周期过程 (idt ISO/IEC 12207:1995)

GB/T 16260.1—2006 软件工程 产品质量 第1部分:质量模型 (ISO/IEC 9126-1:2001,IDT)

GB/T 16260.3—2006 软件工程 产品质量 第3部分:内部度量 (ISO/IEC TR 9126-3:2003, IDT)

GB/T 16260.4—2006 软件工程 产品质量 第4部分:使用质量的度量 (ISO/IEC TR 9126-4:2004,IDT)

GB/T 18491.1—2001 信息技术 软件测量 功能规模测量 第1部分:概念定义 (idt ISO/IEC 14143-1:1998)

GB/T 18905.1—2002 软件工程 产品评价 第1部分:概述 (ISO/IEC 14598-1:1999,IDT)

GB/T 18905.2—2002 软件工程 产品评价 第2部分:策划和管理 (ISO/IEC 14598-2:2000, IDT)

GB/T 18905.3—2002 软件工程 产品评价 第3部分:开发者用的过程 (ISO/IEC 14598-3:2000,IDT)

GB/T 18905.4—2002 软件工程 产品评价 第4部分:需方用的过程 (ISO/IEC 14598-4:1999,IDT)

GB/T 18905.5—2002 软件工程 产品评价 第5部分:评价者用的过程 (ISO/IEC 14598-5:1998,IDT)

GB/T 18905.6—2002 软件工程 产品评价 第6部分:评价模块的文档编制 (ISO/IEC 14598-6:2001,IDT)

ISO 9241-10:1996 使用视觉显示终端(VDT)办公的人类工效学要求 第10部分:对话原则

#### 4 术语和定义

在 GB/T 18905.1—2002 和 GB/T 16260.1—2006 中定义的术语适用于本部分,并且这些术语列在了附录 D 中。

#### 5 符号和缩略语

SQA:软件质量保证(组)

SLCP:软件生存周期过程

#### 6 软件质量度量的使用

GB/T 16260 的第2部分、第3部分和第4部分提出了与第1部分“质量模型”一起使用的一组软件质量度量(外部质量、内部质量和使用质量的度量)的建议。这些部分的用户可以修改已定义的度量,和/或也可以使用未列出的度量。当使用一个已修改的或一个未在各部分中定义的新度量时,用户宜说明这些度量与第1部分中的质量模型或任何其他所用的替代质量模型之间的关系。

GB/T 16260 的用户宜从第1部分中选择用于评价的质量特性和子特性,确定要采用的适当的直接测度和间接测度,确定相关的度量,并以客观的方式解释测量结果。GB/T 16260 的用户也可以从 GB/T 18905 系列标准中选择软件生存周期中的产品质量评价过程。上述这些标准给出了测量、评估和评价软件产品质量的方法,旨在供开发者、需方和独立的评价者使用,特别是那些负责软件产品评价的人员(见图1)。

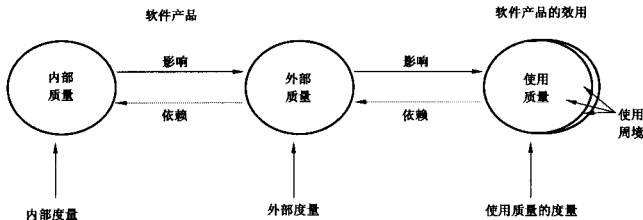


图 1 度量类型之间的关系

内部度量可用于开发阶段的非执行软件产品(例如标书、需求定义、设计规格说明或源代码等)。内部度量为用户提供了测量中间可交付项的质量的能力,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的早期察觉质量问题,并采取纠正措施。

外部度量可以通过测量该软件产品作为其一部分的系统行为来测量软件产品的质量。外部度量只能在生存周期过程中的测试阶段和任何运行阶段使用。在所属系统环境下运行该软件产品即可获得这样的测量。

使用质量的度量是测量产品在特定的使用周境下,满足特定用户达到特定目标所要求的有效性、生产率、安全性和满意度的程度。这只能在真实的系统环境下获得。

用户的质量要求可用使用质量的度量、外部度量甚至是内部度量的质量需求来规定。这些由度量规定的需求宜作为产品评价时的准则。

建议尽可能采用与目标外部度量有密切关系的内部度量,以便能用这些内部度量来预测外部度量的值。然而,往往很难设计出一个能够在内部和外部度量间提供密切关系的严格的理论模型。因此,假设模型可能是模糊的,所以在使用度量时,外部度量和内部度量关系密切程度模型应该使用统计建模的方法。

GB/T 16260 的第 1 部分附录 A 中的 A.4 列出了与有效性和可信性相关的建议和需求。另外,本部分的附录 A 列出了使用度量时的一些考虑细节。

## 7 度量表的阅读和使用

第 8 章列出的度量依据 GB/T 16260.1—2006 中的特性和子特性进行分类。下面是表中的每个度量应给出的信息:

- 度量名称  
在内部度量表和外部度量表中的相应度量的类似名称。
- 度量目的  
在度量应用中以回答问题的形式进行描述。
- 应用的方法  
提供一个应用的大纲。
- 测量、公式和数据元素计算  
给出测量公式,并解释所用的数据元素的意义。  
注:在某些情况下一个度量对应多个公式。
- 测量值解释  
给出范围和最佳值。
- 度量标度类型

度量中使用的标度类型。包括：标称标度、顺序标度、间隔标度、比率标度和绝对标度。

注：附录 C 有详细的解释。

#### g) 测度类型

所用的类型是：规模类型（例如功能规模、源代码规模）、时间类型（例如经时时间、用户时间）、计数类型（变更数、失效数）。

注：附录 C 有详细的解释。

#### h) 测量输入

测量中使用的数据来源。

##### i) 在 GB/T 8566 中的应用

标识出应用度量的软件生存周期过程。

##### j) 目标用户

标识测量结果的使用者。

## 8 度量表

本章列出的度量并不是一个完备集，而且可能还有未经确认的。这些度量按照 GB/T 16260.1—2006 中说明的软件质量特性和子特性的顺序列出。

可用的度量并不仅局限于所列的。其他特定用途的具体度量将由其他相关文档提出，例如：功能规模的测量、精确时间效率的测量。

注 1：推荐使用有关标准、技术报告或指南中规定的度量或测量。功能规模测量的定义见 GB/T 18491.1—2001。

精确时间效率测量的实例见 ISO/IEC 14756。

在一个特定环境中应用度量之前，宜确认该度量（见附录 A）。

### 8.1 功能性度量

外部功能性度量宜对这样的属性进行测量，即包含该软件的系统的功能行为。系统的行为可以从下列方面加以观察：

#### a) 当前实际执行的结果与质量需求规格说明之间的差别；

注 1：功能性质量需求规格说明通常描述为功能需求规格说明。

#### b) 实际用户在操作期间检测到的功能欠缺，这些功能是在规格说明中未明确但却是隐含的需求。

注 2：当隐含的操作或功能被检测出后，宜评审、批准它们，并在规格说明中陈述。就其实现程度达成一致意见。

#### 8.1.1 适合性度量

外部适合性度量（见表 8.1.1）宜对这样的属性进行测量，即在测试和用户运行系统期间出现未满足的功能或不满意的操作。

未满足的功能或不满意的操作可能是：

##### a) 功能或操作未能按照用户手册或需求规格说明中规定的那样执行；

##### b) 功能或操作未能提供合理的和可接受的结果以实现用户任务所期望的特定目标。

#### 8.1.2 准确性度量

外部准确性度量（见表 8.1.2）宜对这样的属性进行测量，即用户遇到不准确的事项的频率。这包括：

##### a) 由于不充分的数据引起的不正确或不精确，如数据的有效数字太少不足以做精确的计算；

##### b) 实际的操作规程与操作手册上描述的规程不一致；

##### c) 在运行期间所执行的任务的实际结果与预期的结果有差别。

#### 8.1.3 互操作性度量

外部互操作性度量（见表 8.1.3）宜对这样的属性进行测量，诸如涉及数据和命令的通信缺失的功



能数或事件数,而这类数据和命令在该软件产品和与之相连的其他系统、其他软件产品或设备之间应易于传送。

#### 8.1.4 安全保密性度量

外部安全保密性度量(见表 8.1.4)宜对这样的属性进行测量,即带有安全保密问题的功能或事件的数目,包括:

- a) 未能防止安全保密输出信息或数据的泄露;
- b) 未能防止重要数据的丢失;
- c) 未能防止非法的访问或非法的操作。

注 1: 建议执行模拟攻击的入侵式测试,因为这种危及安全保密的攻击在通常测试中一般不会发生。真正的安全保密性度量只有在“实际生存系统环境”中,即“使用质量”中才会执行。

注 2: 从独立存在的系统的情况到与互联网相连的系统的情况,对安全保密保障的需求变化很大。确定所需的功能性及确保这些功能的有效性已经在相关标准中广泛阐明。对于那些任何损害造成的影响是重大或是关键的情况,本部分的用户宜使用适当方式和标准来确定安全保密性功能。对于其他情况,用户可以限制其范围为通常接受的“信息技术(IT)”的保护测量,即抗病毒的备份方式及访问权限的控制。

#### 8.1.5 功能性的依从性度量

外部功能性的依从性度量(见表 8.1.5)宜对这样的属性进行测量,即带有依从性问题的功能或事件的数目,这些依从性问题是软件产品不遵从标准、约定、合同或其他法定的需求。

#### 8.2 可靠性度量

外部可靠性度量应对这样一些属性进行测量,在执行测试时,它们与软件作为其一部分的系统的行为有关,以表明在系统运作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。

##### 8.2.1 成熟性度量

外部成熟性度量(见表 8.2.1)应对这样一些属性进行测量,即由于软件本身存在的故障而导致的软件失效的可能程度。

##### 8.2.2 容错性度量

外部容错性度量(见表 8.2.2)是指一旦发生运行故障或违反规定接口时,软件维持规定性能水平的能力的度量。

##### 8.2.3 易恢复性度量

外部易恢复性度量(见表 8.2.3)应对这样一些属性进行测量,即在失效的情况下系统中的软件仍能重新建立适当的性能水平并恢复直接受影响的数据。

##### 8.2.4 可靠性的依从性度量

外部可靠性的依从性度量(见表 8.2.4)应对这样一些属性进行测量,即带有依从性问题的功能或事件的数目,这些问题是指软件产品不遵循与可靠性有关的标准、约定或法规。

#### 8.3 易用性度量

易用性度量是测量软件能被理解、学习和操作,能被吸引,以及遵循易用性法规和指南的程度。

许多外部易用性度量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及宿主系统特性的影响。这并非只是使测量无效,因为评价的软件应当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。(对于通用产品,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,必须使用至少有 8 个用户的样本。在用户进行这些测试时不宜得到任何提示或外来的辅助。

易理解性、易学性及易操作性的度量有两类应用方法:用户测试或产品的使用测试。

注 1: 用户测试

试图使用某一功能的用户测试多种外部度量。在不同个体之间,这些测量的面很广。宜由代表某一指定用户群的样本用户在没有任何提示或外来辅助的情况下执行测试。(对于通用产品,可使用一定范围用户群的代表)。尽管从较小的用户群中也能得到有用的信息,但为了得到可靠的结果,必须使用至少有 8 个用户的

样本。

对所用的测量,宜尽可能建立验收准则或在产品之间进行比较。这意味着测量宜是有已知值的计数项。其结果宜报告均值及均值的标准差。

许多这类度量可用早期的软件原型来测试。所实施的度量将依赖于易用性不同方面的相对重要性,以及后续的使用质量测试的程度。

#### 注2: 产品的使用测试

不同于具体功能的测试,某些外部度量要观察产品在更广泛的应用中对某项功能的使用,以完成作为使用质量测试的一部分的典型任务(GB/T 16260.4—2006)。它的优点是要做的测试较少,但缺点是某些功能在正常使用中很少被用到。

对所用的测量,宜尽可能建立验收准则或在产品之间进行比较。这意味着测量宜是有已知值的计数项。其结果宜报告均值及均值的标准差。

### 8.3.1 易理解性度量

用户宜能选择一个适合他们的使用要求的软件产品。外部易理解性度量(见表 8.3.1)宜能够评估新的用户能否理解下列内容:

- 软件是否合适;
- 怎样用它去完成特殊任务。

### 8.3.2 易学性度量

外部易学性度量(见表 8.3.2)宜能够评估用户要用多长时间才能学会如何使用某一特殊的功能,及评估它的帮助系统和文档的有效性。

易学性与易理解性有很密切的关系,易理解性的测量可作为软件易学性的潜在指标。

### 8.3.3 易操作性度量

外部易操作性度量(见表 8.3.3)宜能够评估用户能否操作和控制软件。易操作性度量可按 ISO 9241-10 中的对话原则进行分类:

- 软件对任务的适合性;
- 软件的自我描述性;
- 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;

对测试功能的选择受使用这一功能的预期频率、功能的关键性以及任何涉及的使用问题的影响。

### 8.3.4 吸引力度量

外部吸引力度量(见表 8.3.4)宜能够评估软件的外观,并受屏幕设计、颜色等因素的影响。这一点对于消费类产品特别重要。

### 8.3.5 易用性的依从性度量

外部易用性的依从性度量(见表 8.3.5)宜能够评估与易用性相关的标准、约定、风格指南或法规的依附性。

## 8.4 效率度量

外部效率度量宜能够测量这样一类属性,即在测试或运行期间包含软件的计算机系统的时间消耗及资源利用特性。在测试和运行的多个实例中,建议调查最大时间和时间分布,因为使用条件对这一测量有很大的影响并会造成波动,例如处理数据的负载、使用频率、相连的站点数等等。因此,效率度量可以包括带误差波动的测量的实际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。

建议列出并调查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。宜建立合适的波动及测量值的有效范围,并把它们与需求规格说明进行比较。

建议标识和定义适合于软件应用的任务,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。

注:

- a) 响应时间:从按动传送键到得到结果为止所需要的时间。这意味着响应时间包括处理时间和传输时间。响应时间只应用于交互系统。在一个单独的系统中,这没有什么明显的不同。但在互联网系统或其他实时系统中,有时传输时间要长一些。
- b) 处理时间:从接受一个消息到送出它的结果之间计算机的经时时间。有时它包括运行的系统开销时间,在其他情况下,它只意味着应用程序所用的时间。
- c) 周转时间:从提出要求到得到结果所需要的时间。在许多情况下,一个周转时间包括多个响应。例如,在银行的自动柜员机的例子中,周转时间是从按动初始键直到取出钱为止所用的时间,这意味着您必须选择交易类型并等待消息、输入口令并等待下一个信息等。

#### 8.4.1 时间特性度量

外部时间特性度量(见表 8.4.1)宜能对这样一些属性进行测量,即在测试及运行中包含软件的计算机系统的时间特性。

#### 8.4.2 资源利用性度量

外部资源利用性度量(见表 8.4.2)宜能对这样一些属性进行测量,即在测试及运行中包含软件的计算机系统的资源利用的特性。

#### 8.4.3 效率的依从性度量

外部效率的依从性度量(见表 8.4.3)宜能对这样一些属性进行测量,即软件产品不能遵循与效率相关的标准、约定或法规的功能数或出现依从性问题的数目。

### 8.5 维护性度量

外部维护性度量宜对这样一些属性进行测量,即在测试和维护中,当软件被维护或修改时,维护者、用户和包含该软件的系统的行为。

#### 8.5.1 易分析性度量

外部易分析性的度量(见表 8.5.1)宜对这样一些属性进行测量,即当试图诊断缺陷或失效的原因,或标识需要修改的部分时,维护者或用户的工作量或耗费的资源。

#### 8.5.2 易改变性度量

外部易改变性度量(见表 8.5.2)宜对这样一些属性进行测量,即当试图实施规定的修改时,通过维护者、用户和包含该软件的系统的行为来测量维护者或用户的工作量。

#### 8.5.3 稳定性度量

外部稳定性度量(见表 8.5.3)宜对修改后的软件进行测试或运行时,与包含该软件的系统意外的行为有关的属性进行测量。

#### 8.5.4 易测试性度量

外部易测试性度量(见表 8.5.4)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,通过维护者、用户和包含该软件的系统的行为来测量维护者或用户的工作量。

#### 8.5.5 维护性的依从性度量

外部维护性的依从性度量(见表 8.5.5)宜对这样一些属性进行测量,即软件产品不遵循所要求的与可维护性相关的标准、约定或法规的功能数和出现依从性问题的数目。

### 8.6 可移植性度量

外部可移植性度量宜对移植活动期间操作者或系统的行为这样一些属性进行测量。

#### 8.6.1 适应性度量

外部适应性度量(见表 8.6.1)宜对这样一些属性进行测量,即系统或用户试图使软件适应于不同的规定环境时的用户行为。当用户必须实施一个新的、而不是原先由适应性要求所指定的适应性规程时,也宜测量适应要求的用户工作量。

#### 8.6.2 易安装性度量

外部易安装性度量(见表 8.6.2)宜对这样一些属性进行测量,即当系统或用户试图在其特定的环境中安装该软件产品时的用户行为。

#### 8.6.3 共存性度量

外部共存性度量(见表 8.6.3)宜对这样一些属性进行测量,即当系统或用户试图将软件与其他的独立软件在公共环境中共享公共资源时的用户行为。

#### 8.6.4 易替换性度量

外部易替换性度量(见表 8.6.4)宜对这样一些属性进行测量,即当系统或用户试图用该软件代替软件环境中其他规定的软件时的用户行为。

#### 8.6.5 可移植性的依从性度量

外部可移植性的依从性的度量(见表 8.6.5)宜对这样一些属性进行测量,即软件产品不遵循所要求的与可移植性相关的标准、约定或法规的功能数和出现依从性问题的数目。

表 8.1.1 适合性度量

外部适合性度量			度量公式及数据元素计算	测量解释	度量标准类型	测度类型	测量输入	在 GB/T 8566 中的应用	目标用户
度量的名称	度量的目的	应用的方法	度量公式及数据元素计算	测量解释	度量标准类型	测度类型	测量输入	在 GB/T 8566 中的应用	目标用户
功能的充分性	被评价的功能其充分程度如何?	适于执行特定任务的功能数与评价的功能数相比较。	$X = 1 - A/B$ A=在评价中检测出有问题的功能数 B=被评价的功能数	$0.0 \leq X \leq 1.0$ 越接近 1.0, 越充分。	绝对标准	X=计数/计数 A=计数 B=计数	需求规格说明 需求规格报告 评价报告	6.5 确认 6.3 质量保证 5.3.9 合格性 测试	开发者 SQA
功能的完整性	按照需求规格说明,功能实现的完整程度如何?	按照需求规格说明对系统做功能性测试(黑盒测试)。对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中描述的功能数相比较。	$X = 1 - A/B$ A=在评价中检测出缺少的功能数 B=需求规格说明中描述的功能数	$0.0 \leq X \leq 1.0$ 越接近 1.0, 越完整。	绝对标准	A=计数 B=计数 X=计数/计数	需求规格说明 需求规格报告 评价报告	6.5 确认 6.3 质量保证 5.3.9 合格性 测试	开发者 SQA
<p>注 1: 测量过程的输入是更新过的需求规格说明。在生存周期中标识的对需求规格说明的任何变更都必须在应用测量过程之前完成。</p> <p>注 2: 建议本度量作为试验性使用。</p> <p>注: 任何缺少的功能均不能通过测试来检查,因为它们并没有实现。为了检测出缺少的功能,建议对需求规格说明中说明的每种功能都要在功能测试时——进行测试,并把它的结果作为“功能实现的完整性”度量的输入。为了检测出已实现,但却不够充分的功能,建议对每种功能都用多个规定的任务进行测试。其结果作为“功能的充分性”度量的输入。因此建议度量的用户在功能测试中同时使用上述两种度量。</p>									
功能实现的覆盖率	功能实现的正确程度如何?	按照需求规格说明对系统做功能测试(黑盒测试)。对在评价中检测出不能正确实现或缺少的功能数进行计数,将其与需求规格说明中描述的功能数相比较。	$X = 1 - A/B$ A=在评价中检测出的不能正确实现或缺少的功能数 B=需求规格说明中描述的功能数	$0.0 \leq X \leq 1.0$ 越接近 1.0, 越好。	绝对标准	A=计数 B=计数 X=计数/计数	需求规格说明 需求规格报告 评价报告	6.5 确认 6.3 质量保证 5.3.9 合格性 测试	开发者 SQA
<p>注 1: 测量过程的输入是变更过的需求规格说明。在生存周期中标识的对需求规格说明的任何修改都必须在应用测量过程之前完成。</p> <p>注 2: 本测量用二进制值检测表示确定特征的存在。</p>									

表 8.1.1 (续)

外部适合性度量					
度量名称	度量目的	应用的方法	测量公式及数据元素计算	度量值解释	度量标准类型
功能规格说明的稳定性(评定)	在进入运行之后,功能规格说明的稳定性如何?	对系统在投入运行后,必须变更功能规格说明中描述的功能数进行计数,将其与需求规格说明中描述的功能总数相比较。	$X=1-A/B$ A=从系统投入运行开始到运行后变更的功能数 B=需求规格说明中描述的功能数	$0.0 \leq X \leq 1.0$ 越接近 1.0, 越好。	绝对标准 A=计数 B=计数 X=计数/计数
注:建议本度量作为试验性使用。					
在 GB/T 8566 中的应用					
目标用户	维护者	需求规格说明	需求规格说明	评价报告	5.4 运作

表 8.1.2 准确性度量

外部准确性度量					
度量名称	度量目的	应用的方法	测量公式及数据元素计算	度量值解释	度量标准类型
预期的准确性	实际的与预期合理的结果之间的差别可接受吗?	执行输入与输出对应的测试用例,把输出与预期合理的结果进行比较。	$X=A/T$ A=按照预期合理的结果,用户得到的结果超出可允许范围的差别的用例数 T=运行时间	$0 \leq X$ 越接近 0 越好。	比率标准 A=计数 T=时间 X=计数/时间
注:预期合理的结果可以在需求规格说明、用户操作手册或用户的期望中标识。					
计算的准确性	最终用户得到不准确结果的频率如何?	记录基于规格说明计算出不准确结果的次数。	$X=A/T$ A=用户得到的不准确计算的次数 T=运行时间	$0 \leq X$ 越接近 0 越好。	比率标准 A=计数 T=时间 X=计数/时间
在 GB/T 8566 中的应用					
目标用户	开发者	需求规格说明	需求规格说明	测试报告	6.5 确认

表 8.1.2 (续)

外部准确性度量					
度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	测量值解释	度量标准 类型
精度	最终用户得到精度不足结果的频率如何?	记录精度不足的结果的次数。	$X = A/T$ A=用户得到的结果精度 不同于要求的精度的次数 T=运行时间	$0 \leq X$ 越接近 0 越好。	比率标准 A=计数 T=时间 X=计数/时间
注: 用来计算外部度量的数据元素设计为使用外部可访问的信息, 因为这对于最终用户、操作者、维护者或需求方使用外部度量, 因此在此外部度量中常出现时间基准度量, 这点与内部度量不同。					

表 8.1.3 互操作性度量

外部互操作性度量					
度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	测量值解释	度量标准 类型
数据的可交换性 (基于数据格式)	对于规定的数据传输, 交换接口的功能已经实现的正确程度如何?	按照数据字段规格说明, 测试系统的每一个下游接口功能输出记录的格式。 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据进行计数, 并与数据交换的总数相比较。	$X = A/B$ A=在测试数据交换时, 证实成功地与其他软件或系统进行交换的数据格式的数目 B=要被交换的数据格式的总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准 A=计数 B=计数 X=计数/计数
数据的可交换性 (基于用户的功能尝试)	最终用户不能在目标软件之间交换数据的程度如何? 在目标软件与其他软件之间数据成功传送的程度如何? 用户能否经常成功地交换数据?	对使用接口功能和失败的次数进行计数。 在目标软件之间数据成功传送的程度如何? 用户能否经常成功地交换数据?	a) $X = 1 - A/B$ A=用户不能与其他软件或系统交换数据的总数据目 B=用户企图交换数据的总数据目 b) $Y = A/T$ T=运行时间的周期	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。  $0 \leq Y$ 越接近 0 越好。	a) 绝对标准 A=计数 B=计数 X=计数/计数  b) 比率标准 Y=计数/时间 T=时间
注: 建议测试规定的数据处理。					

表 8.1.4 安全保密性度量

外部安全保密性度量		度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	度量类型	测量输入	在 GB/T 8566 中的应用	目标用户
访问的可控性	对系统访问的不可控程度如何?	审核性	审核追踪用户访问系统和数据的完整程度如何?	评价在系统的访问历史数据库中记录的访问问题。	$X = A/B$ $A =$ 在访问历史数据库中记录的“用户访问系统和数据”的次数 $B =$ 在评价过程中记录的“用户访问系统和数据”的次数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	$A =$ 计数 $B =$ 计数 $X =$ 计数/计数	测试规格说明 测试报告 测试报告	6.5 确认	开发者
访问的可控性	对检测出的非法操作进行控制程度如何?			对检测出的非法操作的数量进行计数, 将其与规格说明中的非法操作数相比较。	$X = A/B$ $A =$ 检测出非法操作的数量 $B =$ 规格说明中非法操作的类型数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	$A =$ 计数 $B =$ 计数 $X =$ 计数/计数	测试规格说明 测试报告 运行报告	6.5 确认 6.3 质量保证	开发者

注 1: 对数据的访问可能仅在测试活动中测量。

注 2: 建议本度量作为试验性使用。

注 3: 建议执行模拟攻击事件的人侵式测试, 因为这种危及安全保密性的攻击在通常测试中一般不会发生。真正的安全保密性度量只有在“实际生存系统环境”中, 即“使用质量”中才会采用。

注 4: “用户访问系统和数据”的记录可包括防止病毒的“病毒检测记录”。计算机抗病毒概念的目的是创建适当的安全保密措施, 以防止计算机病毒侵入系统或尽早地将其检测出来。

注 1: 如果有必要补充对未预期的非法操作的检测, 那么, 宜进行额外的强化异常操作的测试。

注 2: 建议执行模拟攻击事件的人侵式试验, 因为这种危及安全保密性的攻击在通常测试中一般不会发生。真正的安全保密性度量只有在“实际生存系统环境”(“使用质量”)中才会采用。

注 3: 这些功能能防止非授权的人创建、删除或修改程序或信息。因此, 建议在测试用例中包含这类非法操作。



表 8.1.4 (续)

外部安全保密性度量									
度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	度量类型	测量输入	在 GB/T 8566 中的应用	目标用户
防止数据泄误	数据泄误事件发生的频率如何?	对重大和次要的数据泄误事件发生的次数进行计数。	a) $X = 1 - A/N$ A=重大数据泄误事件发生的次数 N=试图造成数据泄误事件的测试用例数 b) $Y = 1 - B/N$ B=次要数据泄误事件发生的次数 c) $Z = A/T$ 或 $B/T$ T=运行时间的周期(在运行测试期间)	0.0≤X≤1.0 越接近 1.0 越好。 0.0≤Y≤1.0 越接近 1.0 越好。 0≤Z 越接近 0 越好。	a) 绝对 标度 b) 绝对 标度 c) 比率 标度	A=计数 B=计数 N=计数 X=计数/计数 Y=计数/计数 T=时间 Z=计数/时间	测试规格说明 测试报告 运行报告	6.5 确认 5.3.9 合格性测试 5.4 运作	维护者 开发者
<p>注 1: 需要进行强化的异常操作测试来获得重大的和次要的数据泄误事件;</p> <p>注 2: 建议对数据泄误事件的影响分级,例如可分成: 重大的(致命的)数据泄误事件; ——不可再现与不可恢复的事件; ——次级影响散布过宽的事件; ——数据本身的重要性事件。 次要的数据泄误事件: ——可能再现或恢复的事件; ——次级影响没有散布的事件; ——数据本身不够重要的事件。</p> <p>注 3: 用来计算外部度量的数据元素设计为使用外部可访问的信息,因为这有助于最终用户、操作者、维护者或需方便使用外部度量。因此,这里使用的事件和时间的计数不同于内部度量中的计数。</p> <p>注 4: 建议执行模拟攻击事件的人侵式测试,因为这种危及安全保密性的攻击在通常测试中一般不会发生。真正的安全保密性度量只有在“实际生存系统环境”中,也即“使用度量”才会采用。</p> <p>注 5: 建议试验性的使用该度量。</p> <p>注 6: 数据备份是防止数据泄误的有效方法之一。备份的建立确保必要的数据在部分操作数据丢失的事件发生后能很快地恢复。但在本部分中,备份只代表了可靠性度量的一部分。</p> <p>注 7: 建议试验性的使用该度量。</p>									

注 1: 需要进行强化的异常操作测试来获得重大的和次要的数据泄误事件;

注 2: 建议对数据泄误事件的影响分级, 例如可分成:

重大的(致命的)数据泄误事件;

——不可再现与不可恢复的事件;

——次级影响散布过量的事件;

——数据本身的重要性事件。

次要的数据泄误事件:

——可能再现或恢复的事件;

——次级影响没有散布的事件;

——数据本身不够重要的事件。

注 3: 用来计算外部度量的数据元设计为使用外部可访问的信息, 因为这有助于最终用户、操作者、维护者或需方使用外部度量。因此, 这里使用的事件和时间的计数不同于内部度量中的计数。

注 4: 建议执行模拟攻击事件的人侵式测试, 因为这种危及安全保密性的攻击在通常测试中一般不会发生。真正的安全保密性度量只有在“实际生存系统环境”中, 也即“使用度量”才会采用。

注 5: 建议试验性的使用该度量。

注 6: 数据备份是防止数据泄误的有效方法之一。备份的建立确保必要的数据在部分操作数据丢失的事件发生后能很快地恢复。但在本部分中, 备份只代表了可靠性度量的一部分。

注 7: 建议试验性的使用该度量。

表 8.1.5 功能性的依从性度量

度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	测度类型	测量输入	在 GB/T 8566 中的应用	目标用户
功能性的依从性	产品的功能遵循所实施法规、标准和约定的程度如何?	对已满足所要求的依从性的项数进行计数,并与规格说明中要求的依从性项数相比较; 按照依从性的项目来设计测试用例; 对这些测试用例进行功能性测试; 对满足依从性的项数进行计数。	$X=1-A/B$ A=在测试中还未实现规定的功能性的依从性项数 B=规定的功能性的依从性项的总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	A=计数 B=计数 X=计数/计数	有关依从性的产品描述(用户手册或规格说明)以及相关标准、约定和法规 测试规格说明和报告	在 GB/T 8566 中的应用	供方用户
注 1: 随时间收集几种测量值可能有助于分析满足依从性项的增长倾向,以及这些项是否彻底地满足依从性; 注 2: 建议对失效的次数进行计数,因为问题检测是有效测试的目标,也适用于计数和记录。									
界面标准的依从性	界面而遵循应用的方法、标准和约定的程度如何?	对满足所要求的依从性的界面数进行计数,并与规格说明中要求的依从性的界面数相比较。	$X=A/B$ A=按规定正确实现的界面数 B=要求依从性的界面总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	A=计数 B=计数 X=计数/计数	有关依从性的产品描述以及相关标准、约定和法规 测试规格说明和报告	6.5 确认	开发者

注: 标准的所有规定属性都必须测试。

表 8.2.1 成熟性度量

外部成熟性度量				在 GB/T 8566 中的应用		
度量名称	度量的目的	应用的方法	测量公式及数据元素计算	度量标准类型	测量输入	目标用户
估计潜在的故障密度	将来可能出现的故障问题有多少?	对一定的试验周期内检测到的故障数进行计数,并用可靠性增长估计模型来预测未来潜在的故障数。	$X = (ABS(A_1 - A_2)) / B$ $(X: \text{估计残存的潜在故障密度})$ $ABS(X) = \text{绝对值}$ $A_1 = \text{在软件产品中预测的潜在故障总数}$ $A_2 = \text{实际已检测到的故障总数}$ $B = \text{产品的规模}$	绝对标准 $0 \leq X$ 取决于测试阶段,在以后的阶段中 X 值越小越好。	测试报告 运行报告 问题报告 X=计数/规模	开发者 测试者 SQA 用户
针对测试用例的失效密度	在一定的试验周期内检测出多少失效?	对检测到的失效个数和执行测试用例的个数进行计数。	$X = A_1 / A_2$ $A_1 = \text{检测到的失效个数}$ $A_2 = \text{执行测试用例的个数}$	绝对标准 $0 \leq X$ 取决于测试阶段,在以后的阶段中 X 值越小越好。	测试报告 运行报告 问题报告 X=计数/计数	开发者 测试者 SQA

注 1: 如果实际检测到的故障比预测的潜在故障的总数还要多,建议再做预测并估计一个更大的数字。估计一个较大的数字旨在预测合理的潜在失效,但并不能使产品变得更好;

注 2: 建议使用几种可靠性增长估计模型,选出最适合的一种,并通过监视检测的故障重新进行预测;

注 3: 对潜在的故障数预测出一个上限数和一个下限数可能会有帮助;

注 4: 在概述特性时有必要将 X 归到[0,1]区间。

注 1: 在测试的早期阶段,它越大越好,相反,在以后的测试阶段或运行阶段,越小越好。建议监控这一随时间变化的测量趋势;

注 2: 本度量取决于测试用例是否充足,宜将测试用例设计为包括适当的测试用例,即正常的、例外的和异常的用例;

注 3: 在概述特性时有必要将 X 归到[0,1]区间。

表 8.2.1 (续)

外部成熟性度量		测量、公式及数据元素计算		测量值解释	度量标准类型	测量类型	测量输入	在 GB/T 8566 中的应用	目标用户
失效解决	有多少失效的条件已经解决?	在一定的试验周期内,对有多少失效在同样的条件下再未出现进行计数。 维护描述所有这些失效的状态的解决问题报告。	周期	$X = A_1 / A_2$ $A_1$ = 解决的失效数 $A_2$ = 实际检测到的失效总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好 好因为解决了更多的失效。	$A_1$ = 计数 $A_2$ = 计数 $X$ = 计数/计数	测试报告 运行(测试)报告	5.3.8 软件集成 5.3.9 合格性测试 5.4 运作	用户 SQA 维护者
注 1: 在使用这一度量时,建议监视其趋势; 注 2: 可以用可靠性增长模型来估计检测到的潜在失效总数,并用与相似软件产品有关的实际历史数据来校正。在这种情况下,可以比较实际的失效数与预测的失效数,测量残存的未解决的失效数。									
故障密度	在一定的试验周期内检测出多少故障?	对检测到的故障个数进行计数并计算密度。	计算	$X = A/B$ $A$ = 检测到的故障数目 $B$ = 产品的规模	$0 \leq X$ 取决于测试阶段,在以后的阶段中 $X$ 值越小越好。	$A$ = 计数 $B$ = 规模 $X$ = 计数/计数	测试报告 运行报告 问题报告	5.3.8 软件集成 5.3.9 合格性测试 5.4 运作 6.3 质量保证	开发者 测试者 SQA

注 1: 在测试的早期阶段,它越大越好,相反,在以后的测试阶段和运行阶段,越小越好。建议监控这一随时间变化的测量趋势;

注 2: 检测到的故障数除以测试用例数表明测试用例的有效性;

注 3: 在概述特性时有必要将  $X$  归到  $[0,1]$  区间;

注 4: 在对故障进行计数时,应注意:

——重复的可能性,因为多个报告中可能包括与其他报告相同的故障;

——除了故障的可能性,因为有些用户或测试者分不清他们的问题是操作的差错,环境的差错还是软件的失效。

表 8.2.1 (续)

外部成熟性度量	度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量标准类型	测量类型	测量输入	在 GB/T 8566 中的应用	目标用户
	故障排除	已经纠正了多少故障?	对在测试阶段已经排除的故障数进行计数,并与已检测到的故障总数及预测的故障数相比较。	a) $X = A_1/A_2$ $A_1$ = 纠正的故障数 $A_2$ = 实际检测到的故障总数 b) $Y = A_1/A_3$ $A_3$ = 在软件产品中预测的潜在故障总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好,因为只留下较少的故障。 $0 \leq Y$ 越接近 1.0 越好,因为只留下较少的故障。	a) 绝对标准 b) 绝对标准	$A_1$ = 计数 $A_2$ = 计数 $A_3$ = 计数 $X$ = 计数/计数 $Y$ = 计数/计数	测试报告 组织的数据库 集成 5.3.9 合格性测试 6.3 质量保证 6.5 确认	5.3.8 软件集成 5.3.9 合格性测试 5.4 运作 5.4.2 运行测试	开发者 SQA 维护者
	平均失败时间 (MTBF)	软件在运行中失效的频率如何?	对在一定运行周期内发生失效的次数进行计数,并计算失效的平均间隔时间。	a) $X = T_1/A$ b) $Y = T_2/A$ $T_1$ = 运行时间 $T_2$ = 相继发生失效之间的时间间隔累计 $A$ = 实际检测出失效的总数(在运行时间内观察到发生的失效)	$0 < X, Y$ 越长越好,因为期望失效之间的间隔时间较长为好。	a) 比率 b) 比率	$A$ = 计数 $T_1$ = 时间 $T_2$ = 时间 $X$ = 时间/计数 $Y$ = 时间/计数	测试报告 运行(测试)报告 报告 $X$ = 时间/计数 $Y$ = 时间/计数	5.3.8 软件集成 5.3.9 合格性测试 5.4 运作 5.4.2 运行测试	维护者 用户

注 1: 进行下列调查可能是有帮助的:

——失效发生之间的时间间隔分布;

——随着间隔运行时间周期的平均时间的变化;

——指出频繁发生失效的功能的分布,以及功能与使用的可信性有关的操作的分布;

注 2: 故障率或风险率的计算可以互换使用;

注 3: 在概述特性时有必要将  $X$  归到  $[0,1]$  区间。

注 1: 建议在一定的时间周期内监视其变化的趋势;

注 2: 可以用可靠性增长模型来估计预测的潜在故障总数,并用与相似软件产品有关的实际历史数据来校正;

注 3: 建议监视估计的故障解决率  $Y$ : 若  $Y > 1$ , 调查其原因是是否因为已经较早地检测出较多故障,或是因为软件产品包含了特别多的故障。否则,若  $Y < 1$ , 调查是否因为软件中包含的故障比一般检测出的故障少,或是因为该测试不足以检测所有可能的故障;注 4: 在概述特性时有必要将  $X$  归到  $[0,1]$  区间;

注 5: 在对故障进行计数时,应注意重复计算的可能,因为许多报告中都包括与其他报告相同的故障。

表 8.2.1 (续)

外部成熟性度量				度量值解释	度量标准类型	测度类型	测量输入	在 GB/T 8566 中的应用	目标用户
度量名称	度量目的	应用的方法	测量、公式及数据元素计算						
测试覆盖率 (规定的用例?)	在测试期间执行过多少要求的测试用例?	对在测试期间执行的测试用例个数进行计数,并与为充分覆盖的测试用例数相比较。	$X=A/B$ A=在测试期间实际执行的表示运行场量的测试用例数 B=按照覆盖需求要执行的测试用例个数	$0.0 \leq X \leq 1.0$ 越接近 1.0 覆盖率越高。	绝对标准	X=计数/计数 A=计数 B=计数	需求规格说明 或用户手册 测试报告 运行报告	5.3.9 合格性 测试 6.3 质量保证 6.5 确认	开发者 测试者 SQA
测试运行场景	测试运行场景	测试运行场景	测试运行场景						
注:测试用例可以根据软件的规模进行规范,即测试覆盖密度 $Y=A/C$ ,这里 C=被测试的产品规模。Y 越大越好,产品规模可以是用户可测量的功能规模。									
测试的成熟性	产品是否很好地测试? (注解:这是产品在将来测试中达到成功的比率。)	对实际执行通过的测试用例个数进行测试,并与每个需求要执行的测试用例总数相比较。	$X=A/B$ A=在测试或运行中通过的测试用例个数 B=覆盖需求所要运行的测试用例个数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	X=计数/计数 A=计数 B=计数	需求规格说明 测试规格说明 或用户手册 测试报告 运行报告	5.3.9 合格性 测试 6.3 质量保证	开发者 测试者 SQA
注 1: 建议使用有效的历史数据,特别是高峰周期的数据作强度测试。也建议确保执行下列类型的测试并顺利地通过它们: ——用户操作的场景测试; ——高峰期的强度测试; ——过载的数据输入测试。									
注 2: 通过的测试用例可根据软件的规模进行规范,即通过的测试用例密度 $Y=A/C$ ,这里 C=要测试的产品规模。Y 越大越好,产品规模可以是用户可测量的功能规模。									

表 8.2.2 容错性度量

外部容错性度量											
度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量标度类型	测度类型	测量输入	在 GB/T 8566 中的应用	目标用户		
避免死机	软件产品引起整个运行环境死机的情况如何?	对与失效数有关的损坏发生的次数进行计数。 若处在运行情况下,分析用户的运行历史日志。	$X = 1 - A/B$ A=死机发生的次数 B=软件失效的数目	$0.0 \leq X \leq 1.0$ X 值越接近 1.0 越好。	绝对标度	A=计数 B=计数 X=计数/计数	测试报告 运行报告	5.3.8 软件集成 5.3.9 合格性测试 5.4 运作	用户 维护者		
<p>注 1: 死机意味着在系统重新启动之前,用户所有的任务均已暂停,或对系统失去控制,系统只得被迫停机;</p> <p>注 2: 若没有失效或只观察到很少的失效,两次死机之间的间隔时间可能是更合适。</p>											
避免失效	能控制多少种故障模式以避免关键性的或严重的失效?	对已经避免的故障模式的个数进行计数,并与考虑到故障模式数相比较。	$X = A/B$ A=对应故障模式的测试用例,避免关键和严重失效发生的次数 B=在测试中执行的故障模式的测试用例(几乎会引起失效)的个数	$0.0 \leq X \leq 1.0$ X 值越接近 1.0 越好。因为用户能更经常地避免关键或严重的失效。	绝对标度	A=计数 B=计数 X=计数/计数	测试报告 运行报告	5.3.8 软件集成 5.3.9 合格性测试 5.4 运作 6.5 确认	用户 维护者		
<p>注 1: 建议将避免失效的水平依对故障影响的缓急程度进行分类,例如:</p> <ul style="list-style-type: none"><li>—— 关键的:整个系统中止或严重破坏数据库;</li><li>—— 严重的:重要的功能不能执行,并且没有替代的运行方式(与工作有关);</li><li>—— 平均的:绝大部分功能仍然可用,但有限的功能受到限制或要采用替代方式(与工作有关);</li><li>—— 较轻的:少数功能在有限的操作中受到限制;</li><li>—— 忽略的:影响未波及到最终用户。</li></ul> <p>注 2: 避免失效的水平也可根据 GB/T 18492—2001《信息技术 系统及软件完整性级别》中提出的后果的严重性及发生的频率组成的风险矩阵。</p> <p>注 3: 故障模式举例:</p> <ul style="list-style-type: none"><li>——数据超范围;</li><li>——死锁。</li></ul> <p>故障树分析技术可用于检测故障模式。</p> <p>注 4: 测试用例可包括人为的误操作。</p>											

表 8.2.2 (续)

外部容错性度量					
度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	测量值解释	度量标度 类型
抵御错误操作	实现了多少种抵御错误操作能力的功能?	对避免引起关键或严重的失败的误操作的测试用例个数进行计数,并与执行的考虑误操作模式的测试用例个数相比较。	$X = A/B$ $A = \text{避免关键和严重失败发生的次数}$ $B = \text{在测试中执行的(几乎引起失败的)误操作模式的测试用例的个数}$	$0.0 \leq X \leq 1.0$ $X$ 值越接近 1.0 越好,以避免更多的用户误操作。	绝对标度 $A = \text{计数}$ $B = \text{计数}$ $X = \text{计数/计数}$
在 GB/T 8566 中的应用					
				测试报告	5.3.8 软件集成
				运行报告	5.3.9 合格性测试
					5.4 运作
目标用户					
用户维护者					

注 1: 除系统失败之外还会有数据损害。

注 2: 误操作的模式:

——错误的数据库作参数;

——错误的输入数据序列;

——错误的操作序列。

注 3: 故障树分析技术可用于检测误操作模式。

注 4: 本度量可作为经验参数使用。

表 8.2.3 易恢复性度量

外部易恢复性度量					
度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	测量值解释	度量标度 类型
可用性	在规定的时间内周期内系统的可用程度如何?	在类似于生产环境的特定时间周期内执行所有的用户操作来测试系统。 测量试验中系统因故障不能使用后,每次的修复时间。 计算修复的平均时间。	a) $X = (T_c / (T_c + T_r))$ b) $Y = A_i / A_0$ $T_c = \text{操作时间}$ $T_r = \text{修复时间}$ $A_i = \text{用户试图使用软件时能成功使用该软件的总次数}$ $A_0 = \text{观察期间用户试图使用该软件的总次数}$ 这出自于用户可用功能操作的角度。	$0.0 \leq X \leq 1.0$ $X$ 值越大越接近 1.0 越好。因为用户能使用此软件的时间更长。 $0.0 \leq Y \leq 1.0$ $Y$ 值越大越接近 1.0 越好。	(a), (b) 绝对标度 $T_c = \text{时间}$ $T_r = \text{时间}$ $X = \text{时间/时间}$ $A_i = \text{计数}$ $A_0 = \text{计数}$ $Y = \text{计数/计数}$
在 GB/T 8566 中的应用					
				测试报告	5.3.8 软件集成
				运行报告	5.3.9 合格性测试
					5.4 运作
目标用户					
用户维护者					

注: 建议本度量只包括软件提供的自动修复,排除人工执行的维护工作。



表 8.2.3 (续)

外部易恢复性度量				测量、公式及数据元素计算	测量值解释	度量标准类型	测量类型	测量输入	在 GB/T 8566 中的应用	目标用户
度量名称	度量目的	应用的方法	度量的试验周期	$X = T/N$ T=总的宕机时间 N=所观察到的中断次数 宜测量最坏的情况及宕机时间的分布情况。	0<X X 值越小越好,这样系统宕机时间就越短。	比率标准	T=时间 N=计数 X=时间/计数	测试报告 运行报告	5.3.8 软件集成 5.3.9 合格性 测试 5.4 运作 6.5 确认	用户 维护者
平均宕机时间	出现失效时系统逐步启动之前,不能使用系统的平均时间是多少?	测量在特定的试验周期内每次系统不能启动时的宕机时间,计算其平均时间。	从初始的部分恢复到完全恢复系统要花费的平均时间?	注 1: 建议这种可恢复性度量只包括软件提供的自动修复,排除人工执行的维护工作; 注 2: 在概述特性时有必要将 X 归到 [0,1] 区间。						
平均恢复时间	从初始的部分恢复到完全恢复系统要花费的平均时间?	测量在特定的试验周期内每次系统不能启动时的全部恢复时间,计算其平均时间。	测量系统每次宕机所花费的全部恢复时间,计算其平均时间。	注 1: 建议测量多个用例中最坏情况下的最大恢复时间或恢复时间的分布情况; 注 2: 建议这种易恢复性度量只包括软件提供的自动修复,排除人工执行的维护工作; 注 3: 建议区分恢复工作的难度,例如恢复被破坏的数据库比恢复被破坏的事务难得多; 注 4: 在概述特性时有必要将 X 归到 [0,1] 区间。	0<X X 值越小越好。	比率标准	T=时间 X=时间/计数 N=计数	测试报告 运行报告	5.3.8 软件集成 5.3.9 合格性 测试 5.4 运作 6.5 确认	用户 维护者
可重新启动性	在要求的时间内系统能重新启动为用度户提供服务的频度如何?	对系统在目标要求的时间内重新启动并非为用户提供服务的次数进行计数,与在规定的试验周期内系统中断后重新启动的总次数相比。	在要求的时间内系统能重新启动为用度户提供服务的频度如何?	注 1: 建议针对不可操作的严重级别估计不同的重新启动时间,例如:数据库损坏、丢失多项事务、丢失单项事务或临时的数据库损坏; 注 2: 建议这种易恢复性度量只包括软件提供的自动修复,排除人工执行的维护工作。	$0.0 \leq X \leq 1.0$ X 值越大越接近 1.0 越好,这样用户更易于重新启动。	绝对标准	A=计数 B=计数 X=计数/计数	测试报告 运行报告	5.3.8 软件集成 5.3.9 合格性 测试 5.4 运作 6.5 确认	用户 维护者

表 8.2.3 (续)

外部易恢复性度量			度量、公式及数据 元素计算	测量值解释	度量标准 类型	测量类型	测量输入	在 GB/T 8566 中的应用	目标用户
易修复性	度量目的	应用的方法	$X=A/B$ 对成功修复的次数进 行计数,并与规格说 明中要求的测试修复 的总次数相比较。 修复需求举例: 数据库检查点;事务 检查点;重做功能;还 原功能等。	$0.0 \leq X \leq 1.0$ X 值越大越接 近 1.0 越好。 这样产品更能 在一定的用例 中修复。	绝对标准	A=计数 B=计数 X=计数/计数	功能规格说明 测试规格说明 或用户手册 测试报告 运行报告	5.3.8 软件集成 5.3.9 合格性 测试 5.4 运作 6.5 确认	用户 维护者
注:建议这种易恢复性度量只包括软件提供的自动修复,排除人工执行的维护工作。									
修复的有 效性	修复能力的有效程 度如何?	对满足目标修复时间 的测试修复的次数进 行计数,与特定目标 时间所要求的修复次 数相比较。	$X=A/B$ A=满足目标修复时间成 功修复的用例数 B=执行的用例数	$0.0 \leq X \leq 1.0$ X 值越大越接 近 1.0 越好。 这样产品的修 复过程更 有效。	绝对标准	A=计数 B=计数 X=计数/计数	测试报告 运行报告	5.3.8 软件集成 5.3.9 合格性 测试 5.4 运作 6.5 确认	用户 维护者
注:建议这种易恢复性度量只包括软件提供的自动修复,排除人工执行的维护工作。									

表 8.2.4 可靠性的依从性度量

外部可靠性的依从性度量		在 GB/T 8566 中的应用				目标用户
度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	测量输入
可靠性的依从性	遵循与产品的可靠性适用的法规、标准和约定的程度如何?	对要求的依从性已经满足的项数,与在规格说明中要求的依从项数相比较。	$X=1-A/B$ A=在测试中规定的可靠性的依从项数 B=规定的可靠性的依从项数总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	产品的依从性描述(用户手册或规格说明)和相关标准、约定或法规 测试规格说明和测试报告

注: 随时间变化收集若干测量的值可能对分析满足依从项的增长趋势及确定它们是否充分满意是有用的。

表 8.3.1 易理解性度量

外部易理解性度量		在 GB/T 8566 中的应用				目标用户
度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	测量输入
描述的完整性	在阅读完产品描述后能理解的功能(或功能的类型)比例是多少?	用问卷调查或观察用户的表现来进行用户测试和与用户交谈。对被充分理解的功能,与产品的功能总数进行比较。	$X=A/B$ A=被理解的功能(或功能的类型)数 B=功能(或功能的类型)总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	用户手册 运行(测试)报告
演示的获得性	在阅读完产品描述后能获得演示的比例是多少?	进行用户测试和观察用户的表现。对能充分演示的功能进行计数,并与要求演示的功能总数相比较。	$X=A/B$ A=被用户成功访问的演示与指导的数目 B=可用的演示与指导的总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	用户手册 运行(测试)报告

注: 表明潜在用户在阅读了产品描述后理解产品的能力。

注: 表明用户能否找到演示和/或指导。

表 8.3.1 (续)

外部易理解性度量										
度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	测量值解释	度量标准 类型	测度类型	测量输入	在 GB/T 8566 中的应用	目标用户	
使用时演示的获得性	用户在运行时无论何时都有实际需要,用户能得到演示或指导的比例是多少?	观察试图观看演示/指导的用户的表现。可通过录像来观察人類的认知活动。	$X=A/B$ A=当用户试图观看演示时,用户能成功看到演示的事例个数 B=在观察期间用户试图观看演示的事例总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	A=计数 B=计数 X=计数/计数	用户手册 运行(测试)报告 用户监视记录 (录像带或活动记录)	5.3.9 合格性 测试 5.4 运作	用户 维护者	
注:表明在使用产品时用户能否设法得到演示和/或指导。										
演示的有效性	在演示或指导之后用户能成功执行功能的比例是多少?	观察试图观看演示/指导的用户的表现。可通过录像来观察人類的认知活动。	$X=A/B$ A=成功执行的功能个数 B=得到的演示与指导的总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	A=计数 B=计数 X=计数/计数	用户手册 运行(测试)报告	5.3.9 合格性 测试 5.4 运作	用户 维护者	
注:表明在联机演示或指导之后,用户能否成功地执行各种功能。										
明显的功能	基于初始的条件,产品的功能(或功能类别)能被用户识别的比例是多少?	用问卷调查或观察用户的表现进行用户测试和与用户交谈。对用户来说是明显的功能数进行计数,并与功能总数相比较。	$X=A/B$ A=用户识别的功能(或功能类别)的个数 B=实际功能(或功能类别)的总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	A=计数 B=计数 X=计数/计数	用户手册 运行(测试)报告	5.3.9 合格性 测试 5.4 运作	用户 维护者	
注:表明用户是否可通过考察界面(例如通过检查菜单)来落实这些功能。										
功能的易理解性	产品中的功能被用户正确地理解的比例是多少?	用问卷调查进行用户测试和与用户交谈。对易被用户理解的界面功能数进行计数,并与用户的可用功能总数相比较。	$X=A/B$ A=目的能被用户正确描述的界面功能数量 B=界面上可用的功能总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	A=计数 B=计数 X=计数/计数	用户手册 运行(测试)报告	5.3.9 合格性 测试 5.4 运作	用户 维护者	
注:表明用户可通过考察界面(例如通过检查菜单)来落实能否理解这些功能。										

表 8.3.1 (续)

外部易理解性度量					
度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量标准类型
易理解的输入和输出	用户能否理解对输入数据的要求是什么? 以及软件系统提供的输出是什么?	用问卷调查和观察用户的表现进行用户测试和与用户交谈, 对被用户理解的输入和输出项进行计数, 并与用户可得到的总项数相比较。	$X=A/B$ A=用户成功理解的输入和输出数据项的数量 B=从界面上能得到的输入和输出数据项总数	绝对标准 $0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	在 GB/T 8566 中的应用 5.3.9 合格性 测试 5.4 运作
注: 表明用户能理解数据输入的形式和能够正确识别输出数据的含义。					

表 8.3.2 易学性度量

外部易学性度量					
度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量标准类型
功能学习	用户要用多长时间学习使用一项功能?	进行用户测试和观察用户的表现。	T=学习正确使用一项功能要用的平均时间	$0 < T$ 越短越好。	在 GB/T 8566 中的应用 5.3.9 合格性 测试 5.4 运作 6.5 确认
注: 本度量通常作为一种经验或验证性的度量来使用。					
在使用中学习执行一项任务的难易性	用户要用多长时间学习如何有效地执行一项任务?	观察用户从开始学习到能有效地执行为止的表现。	T=用户能在较短时间内执行具体任务所用的操作时间的总和。	$0 < T$ 越短越好。	在 GB/T 8566 中的应用 5.3.9 合格性 测试 5.4 运作 6.5 确认
注 1: 所谓短时间建议为预期的用户操作时间。这样的用户操作时间可作为阈值, 例如把用户首次使用的时间的 70% 作为一个公平的比例。 注 2: 工作量也可以用用户-小时为单位替换表示时间。					

表 8.3.2 (续)

外部易学性度量		度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	测度类型	测量输入	在 GB/T 8566 中的应用	目标用户
用户文档和/或帮助机制的有效性	在帮助用户完成工作任务的比例是多少?	用户文档和/或帮助机制的有效性	在帮助用户完成工作任务的比例是多少?	进行用户测试和观察用户的表现。对在访问联机帮助系统和/或文档后能成功完成任务的总数	$X = A/B$ A=在访问联机帮助系统和/或文档后能成功完成任务的总数 B=测试任务的总量	$0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	A=计数 B=计数 X=计数/计数	运行(测试)报告 用户监视记录	5.3.9 合格性测试 5.4 运作 6.5 确认	用户 人机界面 的设计者
注：可能有三项度量：文档的完整性、帮助机制的完整性或所用的帮助机制与文档结合的完整性。											
所用的用户文档和/或帮助机制的有效性	在阅读用户文档或使用帮助系统后，能正确使用功能的比例是多少?	观察用户的表现。对在阅读了文档或使用帮助系统后能正确使用功能的比例是多少?	观察用户的表现。对在阅读了文档或使用帮助系统后能正确使用功能的比例是多少?	X=A/B A=能使用的功能数 B=提供的功能总数	$0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	A=计数 B=计数 X=计数/计数	用户手册 运行(测试)报告 用户监视记录	5.3.9 合格性测试 5.4 运作 6.5 确认	用户 人机界面 的设计者	
注：本度量通常作为一种经验或验证性的度量来使用，而不是其他。											
帮助的有效性	用户能定位找到帮助主题的比例是多少?	进行用户测试和观察用户的表现。对被正确定位找到的联机帮助的任务数	X=A/B A=正确定位找到的联机帮助的任务数 B=要测试的任务总数	$0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	A=计数 B=计数 X=计数/计数	运行(测试)报告 用户监视记录	5.3.9 合格性测试 5.4 运作 6.5 确认	用户 人机界面 的设计者		
帮助频率	用户必须频繁访问帮助学习操作以便完成其工作任务的程度如何?	进行用户测试和观察用户的表现。对用户访问帮助直到完成任务的事例数	X=A A=访问帮助直到完成任务的事例数	$0 \leq X$ 越接近 0 越好。	绝对标准	X=计数 A=计数	运行(测试)报告 用户监视记录	5.3.9 合格性测试 5.4 运作 6.5 确认	用户 人机界面 的设计者		

表 8.3.3 易操作性度量 a) 与操作用户的期望相符

外部易操作性度量 a) 与操作用户的期望相符			
度量名称	度量的目的	应用的方法	测量、公式及数据元素计算
度量名称	度量的目的	应用的方法	测量、公式及数据元素计算
在使用中操作的一致性	用户界面组成的一致性如何?	观察用户的表现并询问他们的意见。	<p>测量值解释</p> <p>0.0 ≤ X ≤ 1.0 越接近 1.0 越好。</p> <p>期望不一致的不可接受的消 息或功能的数目</p> <p>B = 消息或功能的总数</p> <p>b) Y = N/UOT N = 用户发现与他们的期望不一致的不可接受的操 作的数目</p> <p>UOT = 用户操作时间 (在 观察期间内)</p>
目标用户	用户	人机界面 的设计者	<p>在 GB/T 8566 中的应用</p> <p>测量输入</p> <p>运行 (测试) 报告 用户监视记录</p> <p>度量类型</p> <p>A = 计数 B = 计数 X = 计数/计数</p> <p>5.3.9 合格性 测试 5.4 运作 6.5 确认</p>

注 1: 用户的操作经验对认识各种操作模式通常是有帮助的,而这引导了用户的期望;  
注 2: 对于操作的一致性,“输入的可预测性”和“输出的可预测性”都是有效的;  
注 3: 本度量可用于测量“引导操作的难易”和“通信的流畅”。

表 8.3.3 易操作性度量 b) 可控制性

外部易操作性度量 b) 可控制性			
度量名称	度量的目的	应用的方法	测量、公式及数据元素计算
度量名称	度量的目的	应用的方法	测量、公式及数据元素计算
错误的纠正	用户能否容易地纠正任务中的错误?	进行用户测试和观察用户的表现。	<p>测量值解释</p> <p>0 &lt; T<sub>c</sub> - T<sub>i</sub> 越短越好。</p> <p>T<sub>c</sub> = 完成纠正任务执行中的特定类型的错误的时间</p> <p>T<sub>i</sub> = 开始纠正任务执行中的特定类型的错误的时间</p>
目标用户	用户	人机界面 的设计者	<p>在 GB/T 8566 中的应用</p> <p>测量输入</p> <p>运行 (测试) 报告 用户监视记录</p> <p>度量类型</p> <p>T<sub>c</sub>, T<sub>i</sub> = 时间 T = 时间</p> <p>5.3.9 合格性 测试 5.4 运作 6.5 确认</p>

注: 建议本度量的用户在考虑测试用例时规定错误的类型,例如,严重的类型(显示错误或破坏数据)、输入/输出错误的类型(输入正文错误,向数据库输出数据错误,或显示时的图形错误)或错误操作情况类型(互操作的使用或应急操作)。

表 8.3.3 b) (续)

外部易操作性度量 b) 可控制性					
度量名称	度量目的	应用的方法	测量公式及数据元素计算	度量值解释	度量标准类型
使用中的用户能否容易地修复他们的错误或重新执行任务?	观察运行软件的用户的表现。	观察运行软件的用户的表现。	$X = A/UOT$ A = 用户成功撤销其错误操作 UOT = 在观察周期内用户的操作时间	$0 \leq X$ 越大越好。	比率标准 A = 计数 UOT = 时间 X = 计数/时间
用户能否容易地修复他们的输入?	观察运行软件的用户的表现。	观察运行软件的用户的表现。	a) $X = A/B$ A = 在详细说明输入数据之前成功地修改或变更其中输入数据的屏幕数或表格数 B = 在观察用户操作期间用户试图修改或变更其中输入数据的屏幕数或表格数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准 X = 计数/计数 A = 计数 B = 计数

表 8.3.3 易操作性度量 c) 对执行任务的适合性

外部易操作性度量 c) 对执行任务的适合性					
度量名称	度量目的	应用的方法	测量公式及数据元素计算	度量值解释	度量标准类型
使用中默认可用的性能值?	观察运行软件的用户的表现。	观察运行软件的用户的表现。	$X = 1 - A/B$ A = 在短时间内用户未能建立或选择参数值的次数 B = 用户试图建立或选择参数值但不能成功 (因为用户不能提供默认值)	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准 A = 计数 B = 计数 X = 计数/计数

注 1: 建议观察和记录操作者的表现,并决定多长的时间可作为选择参数值“短时间”;

注 2: 在用每个功能测试设置参数时,还能计算可允许的功能的比率;

注 3: 建议对包含参数设置的功能进行测试。



表 8.3.3 易操作性度量 d) 自描述性(可引导性)

外部易操作性度量 a) 自我描述性(可引导性)										
度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	测度类型	测量输入	在 GB/T 8566 中的应用	目标用户	
使用中的消息的可理解性	用户是否容易理解软件系统的消息? 在开始下一步动作之前是否有任何引起用户延缓理解的消息? 用户是否容易记住重要的消息?	观察正在操作软件的用户的表现。	$X = A/UOT$ A = 由于消息比较费解而导致用户长时间停顿或同一操作反复失败的次数 B = 用户的操作时间(观察时段)	$0.0 \leq X$ 越小和越接近 0.0 越好。	比率标准	A = 计数 UOT = 时间 X = 计数/时间	运行(测试)报告 用户监视记录	5.3.9 合格性测试 5.4 运作 6.5 确认	用户 人机界面 的设计者	
注 1: 对消息理解的难易程度用在开始下一个动作之前因用户理解消息而耽误的时间长度来表示。因此,建议观察和记录操作者的表现并决定停顿多长时间看作是“长时段”。										
注 2: 建议尽可能调查下列用户的理解消息问题的原因:										
a) 注意力: 注意力包含用户可成功识别带有信息的重要消息,例如对用户下一步活动的指南、要查看的数据项的名称以及小心操作的警告。										
——当遇到重要消息时,用户是否未曾看见?										
——由于识别了重要的消息,用户能否在操作中避免错误?										
b) 记忆力: 记忆力包含用户回忆带有信息的重要消息,例如对用户下一步活动的指南、要查看的数据项名称以及小心操作的警告。										
——用户能否容易地回忆重要的消息?										
——回忆重要的消息对用户是否有帮助?										
——是否只需要用户回忆起少数重要的消息而不必太多?										
注 3: 当逐个测试消息时,还要计算能理解的消息占总消息数的比率。										
注 4: 在观察多个参加操作测试的用户时,还要计算能理解消息的用户占全体用户的比率。										
自解释的出错信息	用户提议正确的恢复动作占出错条件的比例是多少?	进行用户测试和观察用户的表现。	$X = A/B$ A = 用户提议正确恢复动作的出错条件个数 B = 测试的出错条件总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	X = 计数/计数 A = 计数 B = 计数	运行(测试)报告 用户监视记录	5.3.9 合格性测试 5.4 运作 6.5 确认	用户 人机界面 的设计者	
注: 本度量通常作为一种经验或验证性的度量来使用。										

表 8.3.3.3 易操作性度量 e) 操作的容错性(不受人为错误影响)

外部易操作性度量 e) 操作的容错性(不受人为错误影响)						
度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	在 GB/T 8566 中的应用
在使用中操作错误的易恢复性	用户能否容易地从更坏的情况下恢复?	观察正在操作软件的用户的表现。	$X=1-A/B$ A=(在用户出错或变更后系统没有通知用户有关的风险,未能成功地从困境中恢复的次数) B=用户出错或变更的总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	X=计数/计数 A=计数 B=计数	在 GB/T 8566 中的合格性 5.3.9 测试 5.4 运作 6.5 确认
注:上述子式代表最坏的情况。本度量的用户可结合下列情况考虑:1)用户造成未被软件系统警告的错误个数;2)用户使用成功或不成功地摆脱困境的次数。						
在使用中人为错误操作的时间间隔	用户能否长时间的操作软件而不犯人为错误?	观察正在操作软件的用户的表现。	$X=T/NC$ 在时间 $[T-T_e]$ 中的时刻 c) $T$ =观察期间的操作时段(或用户人为操作错误之间操作时间的累加) $N$ =用户人为操作错误发生的次数	$0.0 < X$ 越高越好。	X=时间/计数 $T$ =时间 $N$ =计数	在 GB/T 8566 中的合格性 5.3.9 测试 5.4 运作 6.5 确认

注 1:通过对下列用户的行为进行计数,可以检测出人为的错误操作:

- a) 简单的人为错误(疏忽);用户在输入数据时犯简单错误的次数;
- b) 故意错误(过失);在观察期间用户由于误解,在同一操作中重复犯一个错误的次数;
- c) 操作犹豫停顿;在观察期间用户由于犹豫造成长时段停顿的次数;

建议此度量的用户分别衡量上述各项。

注 2:操作停顿视隐含用户对操作的犹豫。

它取决于功能、操作的过程、应用领域和用户是否考虑长时段的操作停顿。因此,评价者需要考虑这些因素并确定一个合理的时间阈值。在交互作业中,“长时段”的阈值为 1 min 到 3 min。

表 8.3.3e) (续)

度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	测量输入	在 GB/T 8566 中的应用	目标用户
(用户错误纠正)的可还原性	用户成功地纠正错误的频率如何?	进行用户测试和观察用户的表现。	a) $X = A/B$ A = 用户成功纠正输入错误的次数 B = 试图纠正输入错误的总次数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	a) 绝对标准 b) 计数 X = 计数/计数	运行 (测试) 报告 用户监视记录	5.3.9 合格性测试 5.4 运作 6.5 确认	用户 人机界面的设计者
	用户正确地还原错误的频率如何?	进行用户测试和观察用户的表现。	b) $Y = A/B$ A = 用户成功纠正错误的条件数 B = 测试的条件数 总次数	$0.0 \leq Y \leq 1.0$ 越接近 1.0 越好。	a) 绝对标准 b) 计数 Y = 计数/计数	运行 (测试) 报告 用户监视记录	5.3.9 合格性测试 5.4 运作 6.5 确认	用户 人机界面的设计者

注: 本度量通常作为一种经验或验证性的度量来使用。

注：本度量通常作为一种经验或验证性的度量来使用。

表 8.3.3 易操作性度量 f) 对个性化的适合性

度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标度类型	测量类型	测量输入	在 GB/T 8566 中的应用	目标用户
易用性	用户能否为方便自己容易地定制其操作规程？	进行用户测试和观察用户的发现。	$X = A/B$ A—能成功定制的功能数 B—试图定制的功能总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标度	A=计数 B=计数 X=计数/计数	用户手册 运行(测试)报告 用户监视记录	5.3.9 合格性测试 5.4 运作 6.5 确认	用户 人机界面 的设计者
定制性	指导最终用户的使用能否为防止最终用户出错而容易地定制操作规程？								
定制性	功能能否定制的比例是多少？								

表 8.3.3 D (续)

外部易操作性度量 D 对个性化的适合性										
度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量标准类型	测度类型	测量输入	在 GB/T 8566 中的应用	目标用户	
<p>注 1: 可以测量用户定制失败的比率:</p> <p><math>Y = 1 - (C/D)</math></p> <p>C = 用户未能定制操作的事例数;</p> <p>D = 用户试图为自己操作而定制的操作总数;</p> <p><math>0 \leq Y \leq 1</math>, 越接近 1.0 越好。</p> <p>注 2: 建议考虑下列内容作为定制操作的变量:</p> <p>—— 选择替代的操作, 例如用选择菜单代替输入指令;</p> <p>—— 组合用户的操作过程, 例如把记录和编辑的操作过程进行组合;</p> <p>—— 设置限制性模板操作, 如把编程过程或设置模板作为输入指南。</p> <p>注 3: 本度量通常作为一种经验或验证性的度量来使用。</p>										
操作规程的减少	用户能否为更方便自己减少操作?	对用户在具体操作中的击打数进行计数, 并与定制操作前后的击打数进行比较。	$X = 1 - A/B$ A = 在定制操作后减少的操作规程数 B = 在定制操作前的操作规程数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	X = 计数/计数 A = 计数 B = 计数	运行 (测试) 报告 用户监视记录	5.3.9 合格性测试 5.4 运作 6.5 确认	用户 人机界面 的设计者	
<p>注 1: 建议对不同的用户任务分别取样, 并区分熟练的用户和初学者。</p> <p>注 2: 操作规程的个数可以通过对操作击打数进行计数来代表, 如点击、拖动、触键、触屏等等。</p> <p>注 3: 包括快捷键。</p>										
物理可访问性	能让有身体缺陷的用户访问的功能比例是多少?	进行用户测试和观察用户的表现。	$X = A/B$ A = 能成功访问的功能数 B = 功能的总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	X = 计数/计数 A = 计数 B = 计数	运行 (测试) 报告 用户监视记录	5.3.9 合格性测试 5.4 运作 6.5 确认	用户 人机界面 的设计者	
<p>注: 不可物理访问的例子如无能力使用鼠标的人或盲人。</p>										

表 8.3.4 吸引力度量

外部吸引力度量					
度量名称	度量目的	应用的方法	测量、公式及数据元素计算	度量值解释	度量标准度类型
相互吸引	界面对用户有多大吸引力?	向用户问卷调查。	在用户使用过界面后,用问卷的方式评估界面对用户吸引力。	取决于问卷调查的方法。	绝对标准 计数
界面外观的易定制性	界面元素能定制为用户满意的外观的比例是多少?	进行用户测试和观察用户的表现。	$X=A/B$ A=能定制为用户满意的外观的界面元素数 B=用户希望定制的界面元素的总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准 X=计数/计数 A=计数 B=计数
注:本度量通常作为一种经验或验证性的度量来使用。					
在 GB/T 8566 中的应用					
目标用户	用户	用户	用户	用户	用户
人机界面	人机界面	人机界面	人机界面	人机界面	人机界面
的设计者	的设计者	的设计者	的设计者	的设计者	的设计者
5.4 运作	5.4 运作	5.4 运作	5.4 运作	5.4 运作	5.4 运作
6.5 确认	6.5 确认	6.5 确认	6.5 确认	6.5 确认	6.5 确认

表 8.3.5 易用性的依从性度量

外部易用性的依从性度量					
度量名称	度量目的	应用的方法	测量、公式及数据元素计算	度量值解释	度量标准度类型
易用性的依从性	遵循产品的易用性适用的标准、约定、风格指南或法规的程度如何?	根据与易用性有关的标准、约定、风格指南或法规,规定所要求的依从性项。 按照依从性项设计测试用例。 对这些测试用例进行功能测试。	$X=1-A/B$ A=在测试期间规定的易用性的依从性项还未实现的数目 B=规定的易用性的依从性项的总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准 X=计数/计数 A=计数 B=计数
注:随时间变化收集若干测量的值可能对分析满足依从性项的增长趋势及确定它们是否充分满意是有用的。					
在 GB/T 8566 中的应用					
目标用户	目标用户	目标用户	目标用户	目标用户	目标用户
提供方	提供方	提供方	提供方	提供方	提供方
用户	用户	用户	用户	用户	用户
测试	测试	测试	测试	测试	测试
6.5 确认	6.5 确认	6.5 确认	6.5 确认	6.5 确认	6.5 确认

表 8.4.1 时间特性度量 a) 响应时间

外部时间特性度量 a) 响应时间										
度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	测量值解释	度量标准 类型	测度类型	测量输入	在 GB/T 8566 中的应用	目标用户	
响应时间 (平均时间)	完成一项规定的任务所花费的时间？ 系统响应一项规定的操作之前要花去多长时间？	开始——一项规定的任务。 测量为完成其操作所花费的时间。 保持每次尝试操作的记录。	$T = \langle \text{获得结果的时间} \rangle - \langle \text{完成命令输入的时间} \rangle$	$0 < T$ 越快越好。	比率标准	T=时间	测试报告 显示经时时间的运行报告	5.3 系统/软件集成 5.3.9 合格性测试 5.4 运作 5.5 维护	用户 开发者 维护者 SQA	
注：建议考虑时间带宽并使用统计分析来测量多项任务(测试样本)，不要只测一个任务。										
响应时间 (平均时间)	运算来说，在一个特定计算机系统负载中，从发出请求直到请求完成为止，用户经历的平均等待时间有多长？	执行若干个并发任务的场景。 测量为完成所选择的到 N) 操作而耗费的平均响应时间。 保存每次尝试操作的记录并为每个方案计算平均时间。	$X = T_{\text{mean}} / TX_{\text{mean}}$ $T_{\text{mean}} = \sum (T_i) / N, (i = 1 \text{ 到 } N)$ $TX_{\text{mean}} = \text{需要的平均响应时间}$ $T_i = \text{第 } i \text{ 个评价的响应时间(点)}$ $N = \text{评价的总数(样本点)}$	$0 \leq X$ 越接近并小于 1.0 越好。	绝对标准	$T_{\text{mean}} = \text{时间}$ $TX_{\text{mean}} = \text{时间}$ $T_i = \text{时间}$ $N = \text{计数}$ $X = \text{时间/时间}$	测试报告 显示经时时间的运行报告	5.3 系统/软件集成 5.3.9 合格性测试 5.4 运作 5.5 维护	用户 开发者 维护者 SQA	
注：所要求的平均响应时间可以从所要求的实时处理规格说明、用户对商业要求的期望或对用户反应的观察中导出。可能需要考虑在人类工效学方面的用户感受。										
响应时间 (最坏情况下的响应时间比)	实现某项功能所需的时间是否在规定的限度内？ 在最坏的情况下，用户能否在规定时间内得到响应？ 在最坏的情况下，用户从软件中获得回答的时间能否短到用户容忍的限度内？	校准测试。 模拟使系统达到最大负载的条件，运行应用并监视结果。 在规定的时间内，用户能否在规定时间内得到响应？ 在最坏的情况下，用户从软件中获得回答的时间能否短到用户容忍的限度内？	$X = T_{\text{mean}} / R_{\text{max}}$ $T_{\text{mean}} = \text{MAX}(T_i) (i = 1 \text{ 到 } N)$ $R_{\text{max}} = \text{需要的最大响应时间}$ $\text{MAX}(T_i) = \text{在评价中最长的响应时间}$ $N = \text{评价的次数(样本点)}$ $T_i = \text{第 } i \text{ 次的响应时间(点)}$	$0.0 < X$ 越接近并小于 1.0 越好。	绝对标准	$T_{\text{mean}} = \text{时间}$ $R_{\text{max}} = \text{时间}$ $T_i = \text{时间}$ $N = \text{计数}$ $X = \text{时间/时间}$	测试报告 显示经时时间的运行报告	5.3 系统/软件集成 5.3.9 合格性测试 5.4 运作 5.5 维护	用户 开发者 维护者 SQA	

表 8.4.1 a) (续)

外部时间特性度量 a) 响应时间						
度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	测量值解释	度量标准 类型	测量输入 在 GB/T 8566 中的应用
				注解：可按下述方式计算响应时间的分布： 统计的最大比率： $Y = T_{dev}/R_{max}$ $T_{dev} = T_{mean} + K(DEV)$ $T_{dev}$ 取值为评价中平均响应时间值加减 2 倍或 3 倍的标准 偏差值。 $K$ ：系数(2 或 3) $DEV = \sqrt{QRT(\sum_{i=1}^N (T_i - T_{mean})^2)/(N-1)}$ ( $i=1$ 到 $N$ ) $T_{mean} = \sum(T_i)/N$ ( $i=1$ 到 $N$ ) $T_{mean}$ = 需要的平均响应时间		

表 8.4.1 时间特性度量 b) 吞吐量

外部时间特性度量 b) 吞吐量						
度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量标准类型	在 GB/T 8566 中的应用
吞吐量	有多少个任务能在给定的时间周期内成功地执行？	按照给定的预想优先级校准各个任务；开始几项作业任务；测量完成测量任务而进行的操作所花费的时间；保留每次尝试操作的记录。	$X = A/T$ $A$ = 完成的任务个数 $T$ = 观察的时间段	$0 < X$ 越大越好。	比率标准 $A$ = 计数 $T$ = 时间 $X$ = 计数/时间	5.3 系统/软件集成 5.3.9 合格性测试 5.4 运作 5.5 维护
						目标用户 用户 开发者 维护者 SQA

表 8.4.1 b) (续)

外部时间特性度量 b)吞吐量		测量公式及数据元素计算	测量值解释	度量标度类型	度量类型	测量输入	在 GB/T 8566 中的应用	目标用户
吞吐量 (平均吞吐量)	在一个设定的单位时间内系统能处理的并发任务的平均数是多少? 按照给定的预想优先级批准各个任务; 执行数项并发任务; 测量在给定的流量中完成选定任务所花费的时间。 保留每次尝试操作的记录	$X = X_{\text{max}}/R_{\text{max}}$ $X_{\text{max}} = \sum(X_i)/N$ $R_{\text{max}} =$ 所要求的平均吞吐量 $X_i = A_i/T_i$ $A_i =$ 在第 $i$ 次评价中在设定的时间段内观察到的并发任务个数 $T_i =$ 在第 $i$ 次评价中设定的时间段 $N =$ 评价的次数	$0 < X$ 越大越好。	绝对标度	$X_{\text{max}} =$ 计数 $R_{\text{max}} =$ 计数 $A_i =$ 计数 $T_i =$ 时间 $X_i =$ 计数/时间 $N =$ 计数 $X =$ 计数/计数	测试报告 显示经时间 的运行报告	5.4 运作 5.5 维护	用户 开发者 维护者 SQA
吞吐量 (最坏吞吐量比率)	系统的并发任务数和和处理量即吞吐量的绝对限度是多大? 校准测试。 模拟系统达到最大负载的条件。 进行并发的作业任务并监视其结果。	$X = X_{\text{max}}/R_{\text{max}}$ $X_{\text{max}} = \text{MAX}(X_i)$ ( $i = 1$ 到 $N$ ) $R_{\text{max}} =$ 所需要的最大吞吐量 $\text{MAX}(X_i) =$ 在评价期间工作任务的最大数 $X_i = A_i/T_i$ $A_i =$ 在第 $i$ 次评价中在设定的时间段内观察到的并发任务个数 $T_i =$ 在第 $i$ 次评价中设定的时间段 $N =$ 评价的次数	$0 < X$ 越大越好。	绝对标度	$X_{\text{max}} =$ 计数 $R_{\text{max}} =$ 计数 $A_i =$ 计数 $T_i =$ 时间 $X_i =$ 计数/时间 $N =$ 计数 $X_{\text{max}} =$ 计数 $X =$ 计数/计数	测试报告 显示经时间 的运行报告	5.4 运作 5.5 维护	用户 开发者 维护者 SQA



表 8.4.1 b) (续)

外部时间特性度量 b) 吞吐量	度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	度量标准 类型	测量类型	测量输入	在 GB/T 8566 中的应用	目标用户
<p>注：可按下述的方式计算分布：</p> $Y = X_{\max} / X_{\min}$ $X_{\min} = X_{\max} + K(DEV)$ $X_{\max}$ 取值为评价中平均吞吐量值加 2 倍或 3 倍的标准偏差值。 $K$ ：系数 (2 或 3) $DEV = \sqrt{QRT(\sum (X_i - X_{\max})^2 / (N-1))}$ (i=1 到 N) $X_{\min} = \sum(X_i) / N$									

表 8.4.1 时间特性度量 c) 周转时间

外部时间特性度量 c) 周转时间	度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	度量标准 类型	测量类型	测量输入	在 GB/T 8566 中的应用	目标用户
周转时间	从发出一个指令开始到发出下一条指令开始之间的时间	因此校准这次试验。开始作业任务。测量用户接受输出结果之间的时间	T = 从用户发完要求到用户接受输出结果之间的时间	0 < T	比率标准	T = 时间	测试报告 显示经时间 的运行报告	5.3 系统、软件 集成 5.3.9 合格性 测试 5.4 运作 5.5 维护	用户 开发者 维护者 SQA
<p>注：建议考虑时间带常并使用统计分析来测量多项任务(样本点)，而不要只测一个任务(点)。</p>									
周转时间 (平均周 转时间)	就发任务及系统利用来说，从发出一个指令开始到相关的任务直到其在一个特定的系统负载之内完成，用经历的平均等待时间有多长？	校准测试。通过执行一些并发性任务(样本点)模拟将负载放在系统中的条件下。测量在给定的流量下完成选定任务所花费的时间。保留每次尝试的记录。	$X = T_{\max} / TX_{\max}$ $T_{\max} = \sum(T_i) / N$ (i=1 到 N) $TX_{\max}$ = 所需要的平均周转时间 $T_i$ = 在第 i 次评价中的周转时间(点) $N$ = 评价数(样本点)	0 < X	绝对标准	$T_{\max} = \text{时间}$ $TX_{\max} = \text{时间}$ $T_i = \text{时间}$ $N = \text{计数}$ $X = \text{时间/时间}$	测试报告 显示经时间 的运行报告	5.3 系统、软件 集成 5.3.9 合格性 测试 5.4 运作 5.5 维护	用户 开发者 维护者 SQA

表 8.4.1 c) (续)

外部时间特性度量 c) 周转时间				在 GB/T 8566 中的应用		目标用户	
度量名称	度量目的	应用的方法	测量公式及数据元素计算	度量标准类型	测量输入	测量输出	目标用户
周转时间 (最坏情况下) 周转时间 的比 率)	为完成一项作业任务所需的时间的绝对长度有多长? 在最坏的情况下, 软件系统要多长的时间才能完成所规定的任务?	校准测试。 根据所执行的任务模拟使系统达到最大负载的条件。 运行选择的工作任务并监视结果。	$X = T_{max} / R_{max}$ $T_{max} = \text{MAX}(T_i) \quad (i=1 \text{ 到 } N)$ $R_{max} = \text{所需要的最大周转时间}$ $\text{MAX}(T_i) = \text{在评价中最长的周转时间}$ $N = \text{评价数(样本点)}$ $T_i = \text{第 } i \text{ 次评价中的周转时间(点)}$	绝对标准 $X = \text{时间/时间}$ $T_{max} = \text{时间}$ $R_{max} = \text{时间}$ $T_i = \text{时间}$ $N = \text{计数}$ $T_{\text{点}} = \text{时间}$	测试报告 显示经时间的运行报告	5.4 运作 5.5 维护	用户 开发者 维护者 SQA
<p>注: 可按下述的方式计算分布:</p> <p>统计的最大比率: <math>Y = T_{\text{点}} / R_{max}</math></p> <p><math>T_{\text{点}} = T_{max} + K(\text{DEV})</math></p> <p><math>T_{\text{点}}</math> 取值为评价中平均周转时间值加减 2 倍或 3 倍的标准偏差值。</p> <p>K: 系数(2 或 3)</p> <p><math>\text{DEV} = \text{SQRT}(\sum((T_i - T_{max})^2) / (N-1)) \quad (i=1 \text{ 到 } N)</math></p> <p><math>T_{max} = \sum(T_i) / N, \quad (i=1 \text{ 到 } N)</math></p> <p><math>TX_{max} = \text{所需要的平均周转时间。}</math></p>							
等待时间	用户等待系统响应的时间占任务所用总时间的比例是多少?	执行多项并发任务的场景。 测量完成选择的操作所花费的时间。 保持每次尝试的记录并为每个场景计算平均时间。	$X = T_i / T_s$ $T_i = \text{用于等待的时间}$ $T_s = \text{任务所用的时间}$	绝对标准 $0 \leq X$ 越小越好。	测试报告 显示经时间的运行报告	5.3 系统/软件 集成 5.3.9 合格性 测试 5.4 运作 5.5 维护	用户 开发者 维护者 SQA
<p>注: 若这些任务只是部分完成, 在做比较时宜使用任务效率度量。</p>							

表 8.4.2 资源利用性度量 a) 输入/输出设备资源的利用性

外部资源利用性度量 a) 输入/输出设备资源的利用性				测量、公式及数据 元素计算	测量值解释	度量标准 类型	测度类型	测量输入	在 GB/T 8566 中的应用	目标用户
度量名称	度量目的	应用的方法								
I/O 设备的利用性	I/O 设备利用是否过高,造成效率不高?	执行大量的并发任务,记录 I/O 设备利用情况,并与设计目标进行比较。		$X = A/B$ $A = \text{占用 I/O 设备的时间}$ $B = \text{在设计时规定占用 I/O 设备的时间}$	$0.0 \leq X \leq 1.0$ 越接近并小于 1.0 越好。	绝对标准	A=时间 B=时间 X=时间/时间	测试报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 SQA
I/O 负载的限度	在完成功能时, I/O 利用的绝对限度是什么?	校准测试条件。模拟使系统达到最大 I/O 负载的条件。运行并监视结果。		$X = A_{\max}/R_{\max}$ $A_{\max} = \text{MAX}(A) \text{ (} G=1 \text{ 到 } N \text{)}$ $R_{\max} = \text{要求的 I/O 最大的消息数}$ $\text{MAX}(A) = \text{从第 1 次到第 } N \text{ 次评价中 I/O 消息的最大数}$ $N = \text{评价次数}$	$0.0 \leq X$ 越小越好。	绝对标准	$A_{\max} = \text{计数}$ $R_{\max} = \text{计数}$ $A = \text{计数}$ $N = \text{计数}$ $X = \text{计数/计数}$	测试报告 运行报告显示 经时间	5.3.9 合格性 测试 5.4 运作 5.5 维护	用户 开发者 维护者 SQA
与错误相关的 I/O	用户是否经常遇到与操作有关的 I/O 设备的问题?	校准测试条件。模拟使系统达到最大 I/O 负载的条件。运行并记录由于 I/O 失败造成的错误数及警告次数。		$X = A/T$ $A = \text{警告消息数或系统失效数}$ $T = \text{观察期间用户运行的时间}$	$0.0 \leq X$ 越小越好。	比率标准	A=计数 T=时间 X=计数/时间	测试报告 运行报告显示 经时间	5.3.9 合格性 测试 5.4 运作 5.5 维护	用户 维护者 SQA
平均 I/O 完成比率	在规定的时间内和规定的利用率的情况下,与错误消息和失败相关的 I/O 的平均数是多少?	校准测试条件。模拟系统达到最大负载的条件。运行应用并记录由于 I/O 失败引起的错误数及警告的次数。		$X = A_{\text{消息}}/R_{\text{消息}}$ $A_{\text{消息}} = \sum(A)/N$ $R_{\text{消息}} = \text{所要求的 I/O 消息的平均数}$ $A = \text{在第 } i \text{ 次评价中 I/O 错误消息数}$ $N = \text{评价次数}$	$0.0 \leq X$ 越小越好。	绝对标准	$A_{\text{消息}} = \text{计数}$ $R_{\text{消息}} = \text{计数}$ $A = \text{计数}$ $N = \text{计数}$ $X = \text{计数/计数}$	测试报告 运行报告显示 经时间	5.3.9 合格性 测试 5.4 运作 5.5 维护	用户 开发者 维护者 SQA

表 8.4.2 a) (续)

外部资源利用性度量 a) 输入/输出设备资源的利用性					在 GB/T 8566 中的应用		
度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量类型	测量输入	目标用户
I/O 设备利用的等待时间	在用户的等待时间方面 I/O 设备利用的影响如何?	并行执行大量的任务, 测量用户等待时间作为 I/O 设备操作的结果。	$T = \text{为等待 I/O 设备操作完成所花费的时间}$	$0 < T$ 越小越好。	比率标准 T=时间	测试报告 运行报告	合格性 测试 5.4 运作 5.5 维护 用户 开发者 维护者 SQA

注: 建议调查几次测试与运行实例中的最大时间与时间分布, 因为这一测量往往因使用问题而引起波动。

表 8.4.2 资源利用性度量 b) 内存的资源利用性

外部资源利用性度量 b) 存储器资源的利用性					在 GB/T 8566 中的应用		
度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量类型	测量输入	目标用户
最大的内存利用数	在完成某个功能时, 需要的内存的绝对限度是多少?	校准测试条件。模拟使系统达到最大负载的条件, 运行并监视结果。	$X = A_{\max}/R_{\max}$ $A_{\max} = \text{MAX}(A_i) (i=1 \text{ 到 } N)$ $R_{\max} = \text{所要求的与内存有关的最大错误消息数}$ $\text{MAX}(A_i) = \text{从第 1 次到第 } N \text{ 次评价与内存有关的最大错误消息数}$ $N = \text{评价次数}$	$0.0 \leq X$ 越小越好。	$A_{\max} = \text{计数}$ $R_{\max} = \text{计数}$ $A_i = \text{计数}$ $N = \text{计数}$ $X = \text{计数/计数}$	测试报告 运行报告 运行时间	合格性 测试 5.4 运作 5.5 维护 用户 开发者 维护者 SQA
内存错误的平均发生数	在系统中规定的负载下和规定的时间长度内与内存有关的错误消息及失效的平均次数是多少?	校准测试条件。模拟使系统达到最大负载的条件。运行应用并记录由于内存失效造成的错误数和警告次数。	$X = A_{\max}/R_{\max}$ $A_{\max} = \Sigma(A_i)/N$ $R_{\max} = \text{所要求的与内存有关的平均错误消息数}$ $A_i = \text{在第 } i \text{ 次评价中与内存有关的错误消息数}$ $N = \text{评价次数}$	$0.0 \leq X$ 越小越好。	$A_{\max} = \text{计数}$ $R_{\max} = \text{计数}$ $A_i = \text{计数}$ $N = \text{计数}$ $X = \text{计数/计数}$	测试报告 运行报告 运行时间	合格性 测试 5.4 运作 5.5 维护 用户 开发者 维护者 SQA

表 8.4.2 b) (续)

外部资源利用性度量 b) 存储器资源的利用性	度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	测量值解释	度量标准 类型	测量类型	测量输入	在 GB/T 8566 中的应用	目标用户
	内存错误 与时间的 比率	在设定的时间周期 内及规定的资源利 用率下,经历了多 少次内存错误?	校准测试条件。模拟 便系统达到最大负载 的条件。运行应用并 记录由于内存失效造 成的错误数和警告 次数。	$X = A/T$ $A =$ 警告消息数或系统失 效率 $T =$ 在观察期内用户的操 作时间	$0.0 \leq X$ 越小越好。	比率标准	$X =$ 计数/时间 $A =$ 计数 $T =$ 时间	测试报告 运行报告显示 经时间	5.3.9 合格性 测试 5.4 运作 5.5 维护	用户 维护者 SQA

表 8.4.2 资源利用性度量 c) 传输资源的利用性

外部资源利用性度量 c) 传输资源的利用性	度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	测量值解释	度量标准 类型	测量类型	测量输入	在 GB/T 8566 中的应用	目标用户
	最大的传 输利用 限度	完成一项功能所需 的传输的绝对限度 是什么?	评价对一个要达到最 大负载的系统来说什 么是所需的。模拟这 种条件。运行应用并 监视结果。	$X = A_{max}/R_{max}$ $A_{max} = \text{MAX}(A_i)$ ( $i = 1$ 到 $N$ ) $R_{max} =$ 所要求的与传输有关 的最大错误消息与失效数 $\text{MAX}(A_i) =$ 从第 1 次到第 $N$ 次评价中与传输有关的最 大错误消息和失效的次数 $N =$ 评价次数	$0.0 \leq X$ 越小越好。	绝对标准	$A_{max} =$ 计数 $R_{max} =$ 计数 $A =$ 计数 $N =$ 计数 $X =$ 计数/计数	测试报告 运行报告显示 经时间	5.3.9 合格性 测试 5.4 运作 5.5 维护	用户 开发者 维护者 SQA
	介质设备 利用的 平衡	在设定的时间周期 内,不同介质同步 运行的程度如何?	校准测试条件。模拟 便系统传输达到最大 负载的条件。运行应 用并记录处理不同的 介质类别所造成的 延迟。	$X = \text{SyncTime}/T$ $\text{SyncTime} =$ 在连线的资源 上耗费的时间 $T =$ 期望不同介质同步完 成其任务所要求的时间 周期	越小越好。	比率标准	$\text{SyncTime} =$ 时间 $T =$ 时间 $X =$ 时间/时间	测试报告 运行报告显示 经时间	5.3.9 合格性 测试 5.4 运作 5.5 维护	用户 维护者 SQA

表 8.4.2 c) (续)

表 8.4.3 效率的依从性度量

效率的依从性度量	度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	测量输入	在 GB/T 8566 中的应用	目标用户
效率的依从性	产品的效率遵循于适用的法规、标准及约定的程度如何?	计算所要求的依从性已得到满足的项数,从性的项目的比率) A=在测试期间规定的效率依从性还未实现的项数 B=规定在效率上有依从性要求的项数	观察试图解决失效的用户或维护者的表现。	$X = 1 - A/B$ (X:与效率有关的满意依从性的项目的比率) A=在测试期间规定的效率依从性还未实现的项数 B=规定在效率上有依从性要求的项数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越依从。	绝对标准 A=计数 B=计数 X=计数/计数	产品对依从性的描述(用户手册或规格说明) 6.5 确认和相关的标准、约定或法规 测试规格说明和报告	5.3.9 合格性测试 5.4 操作 5.5 维护	供应者 用户

注:随时收集几种测量结果是有用的,以分析增加满意的依从性项目的趋势和决定它们是否充分满意。

表 8.5.1 易分析性度量

外部易分析性度量	度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	测量输入	在 GB/T 8566 中的应用	目标用户
审核追踪的能力	用户能否标识引起失效的是哪个具体操作? 维护者能否容易地发现哪个具体操作引起失效?	观察试图解决失效的用户或维护者的表现。	$X = A/B$ A=在运行中实际记录到的数据数 B=计划在运行中要记录的数据的足以监视软件状态的数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准 A=计数 B=计数 X=计数/计数	问题解决报告 运行报告 测试 5.4 操作 5.5 维护	问题解决报告 运行报告	5.3.9 合格性测试 5.4 操作 5.5 维护	操作者 开发者 维护者
对诊断功能的支持	在支持原因分析方面诊断功能的能力? 用户能否标识引起失效的是哪个具体功能? (用户可能用替代的操作来避免同样的失效发生。) 维护者能否容易地发现失效的原因?	观察试图用诊断功能解决失效的用户或维护者的表现。	$X = A/B$ A=维护者能(利用诊断功能)进行失效关系的数据数 B=登记的失效总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准 A=计数 B=计数 X=计数/计数	问题解决报告 运行报告 测试 5.4 操作 5.5 维护	问题解决报告 运行报告	5.3.9 合格性测试 5.4 操作 5.5 维护	操作者 开发者 维护者

表 8.5.1 (续)

外部易分析性度量									
度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	测量的类别	测量输入	在 GB/T 8566 中的应用	目标用户
失效分析的能力	用户能否标识引起失效的是那个具体操作? 维护者能否容易地找到失效的原因?	观察试图解决失效的用户或维护者的表现。	$X = 1 - A/B$ A = 原因仍未找到的失效数 B = 登记的失效总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	A = 计数 B = 计数 X = 计数/计数	问题解决报告 运行报告	5.3.9 合格性测试 5.4 运作 5.5 维护	用户 操作者 开发者 维护者
失效分析的效率	用户能否有效地分析失效的原因? (用户有时通过设定参数来进行维护。)	观察试图解决失效的用户或维护者的表现。	$X = \text{Sum}(T)/N$	$0.0 \leq X$ 越短越好。	比率标准	T = 时间	问题解决报告 运行报告	5.3.9 合格性测试 5.4 运作 5.5 维护	操作者 开发者 维护者
状态监视的能力	通过取得运行时的监视数据, 用户是否能标识引起失效的具体操作? 通过取得运行时的监视数据, 维护者是否容易找到失效的原因?	观察用户或维护者试图取得运行时记录运行状态的监视数据时用户或维护者的表现。	$X = 1 - A/B$ A = 维护者(或用户)未能取得监视数据的事例个数 B = 维护者(或用户)企图取得运行时记录软件状态的监视数据的事例个数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	A = 计数 B = 计数 X = 计数/计数	问题解决报告 运行报告	5.3.9 合格性测试 5.4 运作 5.5 维护	用户 操作者 开发者 维护者

注 1: 建议测量在最坏情况下的最长时间和表示偏差的时间长度(带宽)。

注 2: 建议排除在测量时失效原因尚未发现的失效数, 但这类不明原因的失效的比率宜一起测量和提出。

注 3: 从个别用户的观点看, 要关心时间, 但从维护者的观点看, 还要关心付出的工作量。因此人-小时也可以代替时间。

注 1: 建议测量在最坏情况下的最长时间和表示偏差的时间长度(带宽)。

注 2: 建议排除在测量时未发现失效原因尚未发现的失效数, 但这类不明原因的失效的比率宜一起测量和提出。

注 3: 从个别用户的观点看, 要关心时间, 但从维护者的观点看, 还要关心付出的工作量。因此人-小时也可以代替时间。



表 8.5.2 易改变性度量

外部易改变性度量		度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	测量的类别	测量输入	在 GB/T 8566 中的应用	目标用户
变更周期 的效率			用户的问题是否能在可接受的时间限度内得到满意地解决?	监视用户和供方之间的相互关系。 记录从用户最初提出 要求到问题解决为止 的时间。	平均时间: $T_w = \text{Sum} (T_n)/N$ $T_n = T_{in} - T_{out}$ $T_{in}$ = 用户借问题报告完成 向供方发送维护请求的 时间 $T_{out}$ = 用户收到修订版本发 行(或状态报告)的时间 $N$ = 修订版本的次数	$0 < T_w$ 越短越好, 除非 非修订版本的 次数太多。	比率标准	$T_n$ = 时间 $T_{in}, T_{out}$ = 时间 $N$ = 计数 $T_w$ = 时间	问题解决报告 维护报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	用户 维护者 操作者
实施变更 所用的 时间			维护者能否容易地变更软件以解决失效问题?	观察在试图变更软件时用户或维护者的表现。 另外, 调查问题解决 报告或维护报告。	平均时间: $T_w = \text{Sum} (T_n)/N$ $T_n = T_{out} - T_{in}$ $T_{out}$ = 通过变更软件消除 失效原因的时间(或状态 报告返回用户的时间) $T_{in}$ = 找到失效原因的时间 $N$ = 登记并消除了的失 效数	$0 < T_w$ 越短越好, 除非 非失效数 太多。	比率标准	$T_n$ = 时间 $T_{in}, T_{out}$ = 时间 $T_w$ = 时间	问题解决报告 维护报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 操作者

注 1: 建议测量在最坏情况下的最长时间和表示偏差的时间带宽。  
注 2: 建议排除在测量时失效原因尚未发现的失效数, 但这类不明原因的失效的比率宜一起测量和介绍。  
注 3: 从个别用户的观点看, 要关心时间, 但从维护者的观点看, 还要关心付出的工作量, 因此也可以用人-小时代替时间。

表 8.5.2 (续)

外部易变或变化度量		度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型	测量的类别	测量输入	在 GB/T 8566 中的应用	目标用户
修改的复杂性	维护者能否容易地变更软件以解决问题?	观察维护者在试图变更软件时的行为。另外,调查问题解决报告或维护报告和产品描述。	观察维护者在试图变更软件时的行为。另外,调查问题解决报告或维护报告和产品描述。	$T = \text{Sum}(A/B)/N$ A=变更所花的工作时间 B=软件变更的规模 N=变更的次数	0<T 越短越好,或所要求的变更数过多。	比率标准	A=时间 B=规模 N=计数 T=时间	问题解决报告 维护报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 操作者 维护者
	用户或维护者能否容易地变更参数以变更软件并解决问题?	观察用户或维护者在试图变更软件时的表现。另外,调查问题解决报告或维护报告。	观察用户或维护者在试图变更软件时的表现。另外,调查问题解决报告或维护报告。	$X = 1 - A/B$ A=维护者未能用参数变更软件件的次数 B=维护者企图用参数变更软件件的次数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	绝对标准	A=计数 B=计数 X=计数/计数	问题解决报告 维护报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 操作者 用户
软件变更能力的抑制	用户能否容易地识别修订的版本? 维护者能否容易地变更软件以解决问题?	观察用户或维护者在试图变更软件时的表现。另外,调查问题解决报告或维护报告。	观察用户或维护者在试图变更软件时的表现。另外,调查问题解决报告或维护报告。	$X = A/B$ A=实际记录在工作日志上的变更次数 B=计划在工作日志上记录的足以追踪软件变更的次数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好或者越接近 0 说明变更很少发生。	绝对标准	A=计数 B=计数 X=计数/计数	用户手册或规格说明 格说明 问题解决报告 维护报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 操作者 维护者

注:软件的变更规模可以按源程序中可执行语句、需求规格说明中的变更项数或文档变更的页数来计算。

表 8.5.3 稳定性度量

外部稳定性度量				度量标准 类型	测量的类别	测量输入	在 GB/T 8566 中的应用	目标用户
度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	测量值解释	比率标准	问题解决报告 维护报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 操作者 维护者
变更成功 的比率	在维护之后用户能 否运行软件系统不 再出现失败?	观察用户或维护者在 维护后运行软件时的 表现。	$X = N_s / T_s$ $Y = \{(N_s / T_s) / (N_s / T_s)\}$	$0 \leq X, Y$ 越小越接近 0 越好。	$N_s, N_t =$ 计数 $T_s, T_t =$ 时间 $X =$ 计数/时间 $Y = [(X \text{ 计数/时} \text{间}) / (X \text{ 计数/时} \text{间})]$	问题解决报告 维护报告 运行报告	测试 5.4 运作 5.5 维护	开发者 操作者 维护者
维护者能 否容易地 缓解因维 护引起的 失败?	维护者能容易地 缓解因维护的副作 用引起的失败?	对用户和维护者在维 护前后运行软件时遇 到的失败数进行 计数。	$N_s =$ 软件变更后用户在运 行中遇到失败的事例数 $N_t =$ 软件变更前用户在运 行中遇到失败的事例数 $T_s =$ 软件变更后特定观察 期内运行的时间 $T_t =$ 软件变更前特定观察 期内运行的时间	$0 \leq X$ 越小越接近 0 越好。	$A =$ 计数 $N =$ 计数 $X =$ 计数/计数	问题解决报告 运行报告	测试 5.4 运作 5.5 维护	开发者 操作者 维护者
修改影响 局部化 (变更后 出现失 效)	在维护之后用户能 否运行软件系统不 再出现失败? 维护者能容易地 缓解因维护的副作 用引起的失败?	对在变更之后出现受 变更影响并且相互连 锁的失败次数进行 计数。	$X = A / N$ $A =$ 在特定的时期内通过 变更解决失败后再出现的 失败数 $N =$ 解决的失败数	$0 \leq X$ 越小越接近 0 越好。	$A =$ 计数 $N =$ 计数 $X =$ 计数/计数	问题解决报告 运行报告	测试 5.4 运作 5.5 维护	开发者 操作者 维护者

注 1: X 和 Y 隐含“在变更后遇到失败的频率”和“在变更前遇到失败频率的波动”。

注 2: 软件的修订版被用来解决问题时, 用户可能需要特定的一段时间来确定软件变更的副作用。

注 3: 建议比较在变更前后的失败频率。

注 4: 如果已经标识了变更的功能, 建议确定在变更的功能中是否遇到检测出的失败, 它影响的范围可算每个失败来衡量。

注: X 隐含着“每个解决的失败所显现的连锁失败”。建议通过检查当前的失败是否归结于解决原先失败所引起的变更来进行尽可能给出精确的测量。

表 8.5.4 易测试性度量

外部易测试性度量					
度量名称	度量目的	应用的方法	测量公式及数据元素计算	测量值解释	度量标准类型
内置测试功能的有 效性	用户与维护者能否 不必准备附加的测 试设施就可容易地 做运行测试?	观察用户或维护者在 维护后测试软件系统 时的表现。	$X=A/B$ A=维护者能利用合适的 内置测试功能的事例数 B=测试机会的事例数	$0.0 \leq X \leq 1.0$ 越大越接近 1.0越好。	绝对标准
注：内置测试功能的例子包括模拟功能、备用的预检功能等。					
重新测试 的效率	用户与维护者能否 容易地进行运行测 试并确定软件是否 准备好运行?	观察用户或维护者在 维护后测试软件系统 时的表现。	$X=\text{Sum}(T)/N$ T=为确认是否已经解决 所报告的失效而花费的测 试时间 N=解决的失效总数	$0 \leq X$ 越小越好。	比率标准
注：X隐含“在解决失效后做测试的平均时间(工作量)”。如果未能解决或固定失效,则把它们排除在外并分别测量这类失效的比率。					
测试的重 启性	在维护后用户与维 护者能否容易地用 检测点做运行 测试?	观察用户或维护者在 维护后测试软件系统 时的表现。	$X=A/B$ A=在所希望的点上逐步 检测时维护者能够暂停并 重新开始测试的事例数 B=在测试中暂停的总 次数	$0.0 \leq X \leq 1.0$ 越大越接近 1.0越好。	绝对标准
				测量输入	在 GB/T 8566 中的应用
				测量的类别	合格性
				A=计数	5.3.9 合格性
				B=计数	测试
				X=计数/计数	5.4 运作
					5.5 维护
				问题解决报告	5.3.9 合格性
				运行报告	测试
					5.4 运作
					5.5 维护
				问题解决报告	5.3.9 合格性
				运行报告	测试
					5.4 运作
					5.5 维护

表 8.5.5 维护性的依从性度量

外部维护性的依从性度量	度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量标准类型	测量的类别	测量输入	在 GB/T 8566 中的应用	目标用户
维护性的依从性	维护性的依从性	遵循于产品的可维护性适用的法规、标准及约定的程度如何?	对所要求的依从性已得到满足的项数进行计数,并与规格说明中要求依从性的项数相比较。	$X = 1 - A/B$ A=在测试期间规定的可维护性的依从性还未实施的项数 B=规定的维护性的依从性的项目总数	$0.0 \leq X \leq 1.0$ 越接近 1.0 则依从性越好。	绝对标准	A=计数 B=计数 X=计数/计数	产品对依从性的描述(用户手册或规格说明) 6.5 确认和相关的标准、约定或法规 测试规格说明和报告	5.3.9 合格性测试 6.5 确认	供应者 用户

注: 随时时间收集几种测量结果是有用的,以分析增加测量值的依从性项目的趋势和决定它们是否充分满意。

表 8.6.1 适应性度量

外部适应性度量	度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量标准类型	测量的类别	测量输入	在 GB/T 8566 中的应用	目标用户
数据结构的适应性	用户或维护者能否容易地使软件适应新环境中的数据收集?	观察试图使软件适应运行环境时的用户或维护者的表现。	$X = A/B$ A=可操作但因适应性限制引起运行不完善而未察觉到的数据数 B=希望能在软件适应的环境中运行的数据总数	$0.0 \leq X \leq 1.0$ 越大越接近 1.0 越好。	绝对标准	A=计数 B=计数 X=计数/计数	问题解决方案报告 运行报告 X=计数/计数	问题解决方案报告 运行报告 5.4 运作 5.5 维护	5.3.9 合格性测试 5.4 运作 5.5 维护	开发者 维护者 操作者
硬件环境的适应性	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境时的用户或维护者的表现。	$X = 1 - A/B$ A=在结合硬件环境进行运行测试时未能完成任务或不足以使任务满足适当级别的运行功能数 B=进行测试的功能总数	$0.0 \leq X \leq 1.0$ 越大越好。	绝对标准	A=计数 B=计数 X=计数/计数	问题解决方案报告 运行报告 X=计数/计数	问题解决方案报告 运行报告 5.4 运作 5.5 维护	5.3.9 合格性测试 5.4 运作 5.5 维护	开发者 维护者 操作者

注: 这些数据主要包括的数据类型有数据文件、数据库或数据组,以便适应不同的数据量、数据项或数据结构。在公式中的 A 和 B 必须计数相同类别的数据。当业务的范围扩展时,可能需要这样的适应性。

注: 建议进行有硬件环境配置的过载组合测试,这些硬件环境配置可能会与用户的各种运行环境结合起来。

表 8.6.1 (续)

外部适应性度量		度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量标准类型	测度类型	测量输入	在 GB/T 8566 中的应用	目标用户
组织环境 的适应性 (指对组 织的基础 设施的适 应性)	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境的用户或维护者的表现。	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境的用户或维护者的表现。	$X=1-A/B$ A=在用户的业务环境中运行测试期间没有完成任务或不足以使任务满足适当级别的运行的功能总数 B=进行测试的功能总数	$0.0 \leq X \leq 1.0$ 越大越好。	绝对标准	A=计数 B=计数 X=计数/计数	问题解决报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 操作者
	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境的用户或维护者的表现。	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境的用户或维护者的表现。	注 1: 建议在测试中要考虑与用户的业务环境的基础部分相结合的各种可能的问题; 注 2: “组织的环境适应性”涉及用户组织的业务运行环境。“系统软件的环境适应性”涉及及系统的技术运行环境。因而,两者之间的区别是很明显的。				问题解决报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 操作者
	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境的用户或维护者的表现。	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境的用户或维护者的表现。	注 1: 建议在测试中要考虑与用户的业务环境的基础部分相结合的各种可能的问题; 注 2: “组织的环境适应性”涉及用户组织的业务运行环境。“系统软件的环境适应性”涉及及系统的技术运行环境。因而,两者之间的区别是很明显的。				问题解决报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 操作者
系统软件 环境的适 应性 (OS、网 络软件及 合作应用 软件的适 应性)	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境的用户或维护者的表现。	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境的用户或维护者的表现。	$X=1-A/B$ A=在结合操作系统软件或并行应用软件进行运行测试期间未能完成任务或不足以使任务满足适当级别的运行功能数 B=进行测试的功能总数	$0.0 \leq X \leq 1.0$ 越大越好。	绝对标准	A=计数 B=计数 X=计数/计数	问题解决报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 操作者
	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境的用户或维护者的表现。	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境的用户或维护者的表现。	注 1: 建议进行与操作系统软件或并行的应用软件结合的过载组合测试,这些软件可能会在用户的各种运行环境中结合起来运行。 注 2: “组织的环境适应性”涉及用户组织的业务运行环境。“系统软件的环境适应性”涉及及系统技术运行的环境。因而,两者之间的区别是很明显的。				问题解决报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 操作者
	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境的用户或维护者的表现。	用户或维护者能否容易地使软件适应环境?	观察试图使软件适应运行环境的用户或维护者的表现。	注 1: 建议进行与操作系统软件或并行的应用软件结合的过载组合测试,这些软件可能会在用户的各种运行环境中结合起来运行。 注 2: “组织的环境适应性”涉及用户组织的业务运行环境。“系统软件的环境适应性”涉及及系统技术运行的环境。因而,两者之间的区别是很明显的。				问题解决报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 操作者

表 8.6.2 易安装性度量

外部的易安装性度量				度量标准	度量类型	测量输入	在 GB/T 8566 中的应用	目标用户
度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量类型	测量输入	在 GB/T 8566 中的应用	目标用户
易于安装	用户或维护者能否容易地在软件安装运行环境中?	观察试图在运行环境中安装软件时用户或维护者的表现。	$X = A/B$ A = 用户为自己方便成功地变更安装操作的次数 B = 用户为自己方便企图变更安装操作的总次数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	X = 计数/计数 A = 计数 B = 计数	问题解决报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 操作者
注 1: 建议本度量作为试验性使用;								
注 2: 当需要此度量以时间为单位时, 安装所花费的时间是可测量的。								
易于重新安装	用户或维护者能否容易地重新安装软件?	观察试图重新安装软件时用户或维护者的表现。	$X = 1 - A/B$ A = 在安装操作中, 用户试图重新安装而遭到失败的次数 B = 在安装操作中, 用户试图重新安装的总次数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	X = 计数/计数 A = 计数 B = 计数	问题解决报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 操作者
注: 建议本度量作为试验性使用。								
注: 可以使用下列补充性度量。								
1 安装的费力程度: 在安装中用户的手工作数 $X = A/A$ ; A = 为安装所需要用户手工作数;								
2 安装的简易性: 支持安装的级别 $X = A/A$ 是这样一些等级, 如:								
——只要执行安装程序, 其他什么也不必干(最好);								
——按安装指南安装(好);								
——在安装中需要修改程序的源代码(差);								
3 在安装中所费人力的减少, 用户安装过程减少的比率 $X = 1 - A/B$ , 其中:								
A = 在步骤简化后用户必须执行的安装操作步骤数;								
B = 一般要执行的安装操作步骤数;								
4 用户手工安装操作的简易性, 用户手工安装操作的简单程度, $X =$ 用户手工操作简易性的评分;								
简易性的级别举例:								
[非常容易] 只需要用户启动安装功能, 然后观察安装过程;								
[容易] 只需要用户回答安装功能中提出的问题;								
[不容易] 需要用户从表或填充框中看参数;								
[复杂] 需要用户搜寻参数文件, 从中寻找参数, 并对它们变更或写。								
X = 测量值的直接解释。								

表 8.6.3 共存性度量

度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量标准类型	测量类型	测量输入	在 GB/T 8566 中的应用	目标用户
有效的共存性	在与其他软件共同运行时,用户遇到限制或意外失效的频率程度如何?	使用评价的与用户经常使用的其他软件并存的软件。	$X=A/T$ A=在与其他软件共同运行时,用户遇到的限制或意外失效的次数 T=与其他软件共同运行的时间	$0.0 \leq X$ 越接近 0 越好。	比率标准	X=计数/时间 A=计数 T=时间	问题解决报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 SQA 操作者

表 8.6.4 易替换性度量

度量名称	度量目的	应用的方法	测量、公式及数据元素计算	测量值解释	度量标准类型	测量类型	测量输入	在 GB/T 8566 中的应用	目标用户
数据的连续使用	在更换原先的软件之后,观察用户或维护者能否继续使用同样的数据? 软件系统是否成功地转移?	当用户更换软件时,观察用户或维护者的表现。	$X=A/B$ A=在其他更换的软件中使用并证实能继续使用的数据个数 B=在其他更换的软件中使用并计划能继续重新使用的数据个数	$0.0 \leq X \leq 1.0$ 越大越好。	绝对标准	A=计数 B=计数 X=计数/计数	问题解决报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 操作者
功能的内含性	在用软件更换原先软件之后,观察用户或维护者能否继续容易地使用类似功能? 软件系统是否成功地转移?	当用户更换软件时,观察用户或维护者的表现。	$X=A/B$ A=在新版软件中产生类似结果而无需变更的功能数 B=由要更换的其他软件提供的有类似功能并已测试过的功能数	$0.0 \leq X \leq 1.0$ 越大越好。	绝对标准	A=计数 B=计数 X=计数/计数	问题解决报告 运行报告	5.3.9 合格性 测试 5.4 运作 5.5 维护	开发者 维护者 操作者

注:本度量适用于替换整个不同的软件系列的不同版本替换原先版本的情况。

注:本度量适用于更换整个不同的软件系列的不同版本更换原先版本的情况。



表 8.6.4 (续)

外部易替换性度量				度量标准 类型	度量类型	测量输入	在 GB/T 8566 中的应用	目标用户
度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	测量值解释	度量类型	测量输入	在 GB/T 8566 中的应用	目标用户
用户支持 功能的一 致性	新部件与原有用户 界面一致的程度 如何?	观察用户的表现,并 询问他们的意见。	$X = 1 - A_1/A_2$ $A_1$ = 用户发现新功能与自 己期望不一致的不可接 受地数目 $A_2$ = 新功能的总数	$0.0 \leq X$ 越大越好。	绝对标度 $A_1$ = 计数 $A_2$ = 计数 $X$ = 计数/计数	测试报告 运行报告 测试 5.4 运作 6.3 质量保证	5.3.8 软件集成 5.3.9 合格性 测试 5.4 运作 6.3 质量保证	用户 用户界面 设计者 维护者 开发者 测试者 SQA

注 1: 当引入一个不同的软件来代替原先的软件时,这一不同的新软件可当作现有软件的新版本考虑;

注 2: 当新版本中为改进用户界面,变更了交互作用的模式时,建议观察用户的表现并统计用户由于不能与自己从旧版本中得到的印象相符而不能接受新功能的事例数。

表 8.6.5 可移植性的依从性度量

外部可移植性的依从性度量				度量标准 类型	测量的类别	测量输入	在 GB/T 8566 中的应用	目标用户
度量名称	度量目的	应用的方法	测量、公式及数据 元素计算	测量值解释	测量的类别	测量输入	在 GB/T 8566 中的应用	目标用户
可移植性 的依从性	遵循于产品的可移 植性适用的法规、 标准及约定的程度 如何?	对所要求的依从性已 得到满足的项数进行 计数,并与规格说明 中要求依从性的项数 相比较。	$X = 1 - A_1/B$ $A_1$ = 在测试中规定的可移 植性的依从性项数未实现 的项数 $B$ = 规定的可移植性的依 从性的项数	$0.0 \leq X \leq 1.0$ 越接近 1.0 越好。	A = 计数 B = 计数 X = 计数/计数	产品对依从性 的描述(用户手 册或规格说明) 6.5 确认 和相关的标准、 约定及法规。 测试规格说明 和报告	5.3.9 合格性 测试 6.5 确认	供应者 用户

注: 随时间收集的几种测量结果是有用的,以分析增加满意的依从性项目的趋势和决定它们是否充分满意。

附 录 A  
(资料性附录)  
使用度量时的考虑

## A.1 测度的解释

### A.1.1 测试使用周境与运行使用周境之间的潜在差异

在策划使用度量或解释测度时,理解清楚软件所要的使用周境,以及测试使用周境与运行使用周境之间的潜在差异是很重要的。例如:在类似的软件系统中,“学会操作所需的时间”的测度,对技术熟练的操作者与不熟练的操作者常常不一样。下面给出一些潜在差异的例子:

#### a) 测试环境与运行环境之间的差异

测试环境与运行环境之间是否有明显差异?

下面是一些实例:

- 具有较高/相当/较低的运行计算机 CPU 性能的测试环境;
- 具有较高/相当/较低的运行网络和通信性能的测试环境;
- 具有较高/相当/较低的运行操作系统性能的测试环境;
- 具有较高/相当/较低的运行用户界面性能的测试环境。

#### b) 测试的执行与实际运作的执行之间的差异

测试的执行与用户环境中运行的执行之间是否有明显差异?

下面是一些实例:

- 测试环境中功能的覆盖率;
- 测试用例的抽样率;
- 实时事务的自动测试;
- 压力负载;
- 每周 7×24 h(不间断)运行;
- 用来测试异常和差错的合适数据;
- 周期性处理;
- 资源利用率;
- 中断级别;
- 生产强度;
- 干扰。

#### c) 观察用户的特征

测试时用户的特征与运行时用户的特征是否有明显差异?

如下列实例:

- 混合类型的用户;
- 用户的技能水平;
- 专业用户或一般用户;
- 受限用户组或公共用户。

### A.1.2 影响结果有效性的问题

下列问题可能会影响所收集的数据的有效性。

#### a) 收集评价结果的规程

借助工具或设施自动收集/手工收集/问卷调查或面谈。

b) 评价结果的来源

开发者的自述报告/评审者的报告/评价者的报告。

c) 结果数据的确认

开发者自查/由独立评价者检查。

A.1.3 测量资源的平衡

在每个阶段使用的测度的平衡是否适合于评价的目的?

在为内部测量、外部测量和使用质量的测度应用适当的度量范围时,平衡所用的工作量是很重要的。

A.1.4 规格说明的正确性

软件规格说明与实际操作要求之间是否有明显差异?

在不同阶段评价软件产品时,所采取的测量就是与产品的规格说明进行比对。因此,通过确认和验证来确保用于评价的产品规格说明能反映出运行中真实和实际的需要是非常重要的。

A.2 度量的确认

A.2.1 度量的理想性质

为了从质量评价中获得有效结果,度量应具有下列性质。若某种度量不具备这些性质,那么,度量描述应说明对其有效性的约束,并尽可能解释如何处理这类情况。

- a) (度量的)可靠性:可靠性与随机误差有关。如果随机变量不影响度量的结果,则度量是没有随机差错的。
- b) (度量的)可重复性:由相同的评价者使用相同的评价规格说明(包括在相同的环境中)和相同的用户类型及环境,对相同产品重复进行的度量宜在适当的容差范围内得出相同的结果。这里所谓适当的容差应包括诸如疲劳、学习效应等因素。
- c) (度量的)可再现性:由不同的评价者使用相同的评价规格说明(包括在相同的环境中)和相同的用户类型及环境,对相同产品进行的度量宜在适当的容差范围内得出相同的结果。

注1:建议对测量结果的可变性进行统计分析。

- d) (度量的)可用性:度量应明确指出其约束使用条件(如特定因素存在的条件)。
- e) (度量的)指示性:度量标识软件应改进的部分或改进的项,并给出与期望值进行比较的测量结果的能力。

注2:与只检查所需的项目不同,对选定或建议的度量宜提供使用度量可用性的书面证据。

- f) (度量的)正确性:度量应具备下列性质:
  - 1) (测度的)客观性:度量的结果与其数据输入应是有据可查的,即不受评价者、测试用户的感觉或观点的影响(除非满意度或吸引力度量,因为用户的感觉与观点也是测量的对象)。
  - 2) (测度的)公正性:度量不应偏向任何特殊的结果。
  - 3) (测度的)充分精确性:精确性由度量的设计,特别是作为度量基础的材料的选择来确定。度量的用户将描述度量的精确性和灵敏性。
- g) (测度的)意义:测量应产生有关软件行为或质量特性的有意义的结果。度量也应具有成本效益:即成本越高的度量,提供的结果应越具价值。

A.2.2 度量有效性的证实

度量的用户应标识一些证实度量的有效性的方法,例如:

a) 相关性

质量特性值(对运行使用中的主要度量的测度)中的变化可以用度量值中的变化来解释,用线性系数的平方表示。

利用相关性度量,评价者不用直接测量就可以预测质量特性的值。

## b) 跟踪

若度量值  $M$  和质量特性值  $Q$  (对运行使用中的主要度量的测度) 直接相关, 给定一个产品或过程, 当值  $Q(T_1)$  变为  $Q(T_2)$  时, 度量值也以相同的趋势, 从  $M(T_1)$  变为  $M(T_2)$  (若  $Q$  值增加, 则  $M$  值也增加)。

评价者不必直接测量而是通过使用那些具有跟踪能力的度量就可以检测质量特性随时间周期的变化。

## c) 一致性

若质量特性值 (对运行使用中的主要度量的测度)  $Q_1, Q_2, \dots, Q_n$  对应于产品或过程  $1, 2, \dots, n$ , 当有关系  $Q_1 > Q_2 > \dots > Q_n$  时, 则对应的度量值也有关系  $M_1 > M_2 > \dots > M_n$ 。

评价者可以使用具有一致性能力的度量来关注软件的异常部件和易出差错的部件。

## d) 可预测性

若使用时间  $T_1$  的度量来预测时间  $T_2$  质量特性值  $Q$  (对运行使用中的主要度量的测度)。预测误差 = (预测值  $Q(T_2)$  - 实际值  $Q(T_2)$ ) / 实际值  $Q(T_2)$ , 预测误差值会在预测允许的范围内。

评价者可以通过可预测性的度量预测质量特性今后的变化趋势。

## e) 可判别性

度量能够判别出软件质量的高低。

评价者可以使用具有判别能力的度量对软件部件进行分类和对质量特性值定级。

## A.3 使用度量进行估计(判断)与预测(展望)

用如下两种方法在早期阶段估计和预测软件产品的质量特性是最具价值的度量。

## A.3.1 利用当前的数据预测质量特性

## a) 利用回归分析来预测

当通过使用特性(属性)的当前值(数据)来预测同一特性(属性)的未来值(测度)时, 根据一个足够长的时间内观察到的一组数据进行回归分析是有用的。

例如在测试阶段(活动)中获得的 MTBF(平均失效间隔时间)的值可用于估计在运行阶段的 MT-BF。

## b) 利用相关性分析来预测

当用不同属性的当前测量值来预测特性(属性)的未来值(测度)时, 使用一个确认的表明相关性的函数进行相关性分析是有用的。

例如在编码阶段, 模块的复杂性可用来预测在维护过程中程序修改和测试所花费的时间与工作量。

## A.3.2 根据当前的事实估计当前的质量特性

## a) 利用相关性分析来估计

在估计不可直接测量的属性的当前值时, 若任何其他的测度与目标测度密切相关时, 相关性分析方法是有用的。

例如软件产品中遗留的故障数是不可测的, 但它可以用检测出的故障数及故障趋势进行估计。

对于不能直接测量的属性进行预测的那些度量应用下述解释来估计:

- 使用模型来预测属性;
- 使用公式来预测属性;
- 基于经验来预测属性;
- 使用合理判断来预测属性。

对于不能直接测量的属性进行预测的那些度量可以用下述解释来确认:

- 标识要预测的属性的测度;
- 标识要用来预测的度量;

- 进行基于确认的统计分析；
- 将结果归档；
- 定期地重复上述工作。

#### A.4 检测易发生质量问题的部件中的偏差或异常

下列质量控制工具可用来分析在软件产品部件中的偏差和异常情况：

- a) 流程图(软件的功能模块)
- b) 排列分析和排列图
- c) 直方图和散点图
- d) 运行图、相关图和层次图
- e) 鱼骨图
- f) 统计过程控制(软件功能模块)
- g) 检查单

上述工具可用于标识源于数据的质量问题,这些数据是通过应用度量来获得的。

#### A.5 显示测量结果

- a) 显示质量特性评价的结果

对于每个质量特性和子特性可用下列图示法显示质量评价的结果：

雷达图、条形图、数字化的直方图、多变量图、重要性能矩阵图等。

- b) 显示测度

可利用一些有用的图形表示,如排列图、趋势图、直方图、相关图等。

## 附录 B

### (资料性附录)

#### 使用质量的度量、外部度量和内部度量的用法(框架实例)

### B.1 引言

本框架实例是一个高层描述,它描述的是关于如何在软件开发和实现过程中使用 GB/T 16260.1—2006 中的质量模型和相关的度量来获得满足用户要求的质量产品。本实例所示的概念可用不同的定制形式来实现,以适应个体、组织或者项目。本实例使用的 GB/T 8566—2001 中的生存周期过程可作为传统软件开发生存周期的基准,使用的 GB/T 18905.3—2002 中的质量评价过程的步骤可作为传统软件产品质量评价过程的基准。只要能理解基本概念,如果用户愿意,也可以把这些概念映射为其他的软件生存周期模型。

### B.2 开发及质量过程的概述

为了测量可交付项(即使用质量、外部质量和内部质量)的质量,表 B.1 描述了一个示例模型,它将软件开发生存周期过程的活动(从活动 1 到活动 8)与其关键的可交付项及相关的引用模型联系起来。

第一行描述软件开发生存周期过程的活动。(可为适应独特的要求来定制)。第二行描述可能作为测量类别(如使用质量、外部质量或内部质量)的一种实际的测度或者预测。第三行描述可以测量质量的关键可交付项,第四行描述在每个过程活动中可适用于每个可交付项的度量。

表 B.1 质量测量模型

	活动 1	活动 2	活动 3	活动 4	活动 5	活动 6	活动 7	活动 8
阶段	需求分析 (软件与系统)	体系结构设计 (软件与系统)	软件的详细设计	软件编码与测试	软件集成及软件的合格性测试	系统集成及系统合格性测试	软件的安装	软件的验收支持
模型的引用	所需的用户质量 所需的内部质量 所需的外部质量	预测的使用质量, 预测的外部质量, 测量的内部质量	预测的使用质量, 预测的外部质量, 测量的内部质量	预测的使用质量, 测量的外部质量, 预测的外部质量, 质量, 测量的内部质量	预测的使用质量, 测量的外部质量, 预测的外部质量, 质量, 测量的内部质量	预测的使用质量, 测量的外部质量, 测量的内部质量	预测的使用质量, 测量的外部质量, 测量的内部质量	测量的使用质量, 测量的外部质量, 测量的内部质量
活动的关键可交付项	用户的质量需求(规定的), 外部的质量需求(规定的), 内部的质量需求(规定的)	软件/系统体系结构设计	软件详细设计	软件代码,测试结果	软件产品,测试结果	集成的系统,测试结果	安装的系统	交付的软件产品
用于测量的度量	内部度量(外部度量可用于确认需求规格说明)	内部度量	内部度量	内部度量 外部度量	内部度量 外部度量	内部度量 外部度量	内部度量 外部度量	使用质量 度量 内部度量 外部度量

### B.3 质量途径步骤

#### B.3.1 概述

开发周期中对质量的评价分为下列步骤。步骤1必须在需求分析活动中完成,步骤2到步骤5必须在上述定义的每个过程活动中重复进行。

#### B.3.2 步骤1:质量需求的确定

对质量模型中定义的每个质量特性和子特性,用表B.2中的两个例子对每类测量(使用质量、外部和内部质量)确定用户要求的权重。根据分配的相对权重,允许评价者集中精力于最重要的子特性上。

表 B.2 用户要求的特性与权重 a)

使用质量		
	特性	权重(高/中/低)
	有效性	高
	生产率	高
	安全性	低
	满意度	中

表 B.2 用户要求的特性与权重 b)

外部与内部质量		
特性	子特性	权重(高/中/低)
功能性	适合性	高
	准确性	高
	互操作性	低
	安全保密性	低
	功能性的依从性	中
可靠性	成熟性(硬件、软件、数据)	低
	容错性	低
	易恢复性(数据、过程、技术)	高
	可靠性的依从性	高
易用性	易理解性	中
	易学性	低
	易操作性	高
	吸引力	中
	易用性的依从性	高
效率	时间特性	高
	资源利用性	高
	效率的依从性	高

表 B.2 (续)

外部与内部质量		
特性	子特性	权重(高/中/低)
维护性	易分析性	高
	易改变性	中
	稳定性	低
	易测试性	中
	维护性的依从性	高
可移植性	适应性	高
	易安装性	低
	共存性	高
	易替换性	中
	可移植性的依从性	高

注：权重可用高/中/低的方式表示，也可在1~9的范围内用顺序标度来表示(例如1~3=低、4~6=中、7~9=高)。

B.3.3 步骤2:评价的规格说明

每个开发过程活动都要实施本步骤。

质量模型中定义的质量子特性均标识要应用的度量和要求的级别，以便达到在第1步骤中设定的用户要求，并按表B.3的例子加以记录。

对内容阐述的基本输入及用法说明可在表B.1的例子中得到，其中解释了在开发周期的这一阶段中能测量什么。

注：在开发周期的特定活动中，表中的某些行可以是空的，因为在开发过程的早期，不可能测量所有子特性。

表 B.3 质量测量表 a)

使用质量测量类别				
	特性	度量	要求的级别	评估实际结果
	效率			
	生产率			
	安全性			
	满意度			

表 B.3 质量测量表 b)

外部质量测量类别				
特性	子特性	度量	要求的级别	评估实际结果
功能性	适合性			
	准确性			
	互操作性			
	安全保密性			
	功能性的依从性			



表 B.3 (续)

外部质量测量类别				
特性	子特性	度量	要求的级别	评估实际结果
可靠性	成熟性(硬件/软件/数据)			
	容错性			
	易恢复性(数据、过程、技术)			
	可靠性的依从性			
易用性	易理解性			
	易学性			
	易操作性			
	吸引力			
	易用性的依从性			
效率	时间特性			
	资源利用性			
	效率的依从性			
维护性	易分析性			
	易改变性			
	稳定性			
	易测试性			
	维护性的依从性			
可移植性	适应性			
	易安装性			
	共存性			
	易替换性			
	可移植性的依从性			

表 B.3 质量测量表 c)

内部质量测量类别				
特性	子特性	度量	要求的级别	评估实际结果
功能性	适合性			
	准确性			
	互操作性			
	安全保密性			
	功能性的依从性			
可靠性	成熟性(硬件/软件/数据)			
	容错性			
	易恢复性(数据、过程、技术)			
	可靠性的依从性			

表 B.3 (续)

内部质量测量类别				
特性	子特性	度量	要求的级别	评估实际结果
易用性	易理解性			
	易学性			
	易操作性			
	吸引力			
	易用性的依从性			
效率	时间特性			
	资源利用性			
	效率的依从性			
维护性	易分析性			
	易改变性			
	稳定性			
	易测试性			
	维护性的依从性			
可移植性	适应性			
	易安装性			
	共存性			
	易替换性			
	可移植性的依从性			

## B.3.4 步骤3:评价的设计

每个开发过程活动都要实施本步骤。

制定一个包括可交付项的测量计划(类似于表 B.4 中的例子),这些交付项用作要实施的度量和测量过程的输入。

表 B.4 测量计划

子特性	要评价的可交付项	应用的内部度量	应用的外部度量	应用的使用质量度量
1. 适合性	1. 2. 3.	1. 2. 3.	1. 2. 3.	(不适用)
2. 满意度	1. 2. 3.	(不适用)	(不适用)	1. 2. 3.
3.				
4.				
5.				
6.				

## B.3.5 步骤4:评价的执行

每个开发过程活动都要实施本步骤。

执行评价计划,填写表 B.3 例子中的每一列。GB/T 18905 系列标准可用作编制计划和执行测量过程的指南。

#### **B.3.6 步骤 5:反馈给组织**

每个开发过程活动都要实施本步骤。

一旦所有测量均已完成,要把结果映射到表 B.1 中并以报告的形式将结论写成文件。同时标识产品质量需要改进的特定区域以使其满足用户的需要。

## 附录 C

(资料性附录)

## 度量标度类型和测度类型的详细解释

## C.1 度量标度的类型

当度量的用户得到了测量结果并使用该测度进行计算和比较时,宜对每种测度标识下列度量标度类型。对某些测度,平均值、比率或差值可能没有意义。度量标度类型有:标称标度、顺序标度、间隔标度、比率标度和绝对标度。 $M' = F(M)$ ,这里  $F$  是一个容许函数。每个测量标度类型的描述包含容许函数的描述(若  $M$  是一个度量,则  $M' = F(M)$  也是一个度量)。

## a) 标称标度

$M' = F(M)$ ,这里  $F$  是一对一的映射。

标称标度包括分类,例如软件的故障类型(数据、控制、其他)。只有用相同类型的频率计算时,平均值才有意义。只有用经过映射的每种类型的频率计算时,比率也才有意义。因此,平均值和比率可以用来代表相同类型频率的早期和后来情况之间或两个类似情况之间的差。否则,它们可以用来相互比较每种类型各自的频率。

例如:城市交通线标识号、编译器出错消息标识号。

含义说明:只是不同类别的数。

## b) 顺序标度

$M' = F(M)$ ,这里  $F$  是一个单调递增的映射,即:若  $M(x) \geq M(y)$ ,则  $M'(x) \geq M'(y)$ 。

顺序标度包括排序,例如:软件的失效按严重程度排序(忽略不计的、轻微的、严重的、灾难性的)。只有用经过映射的相同顺序的频率计算时,平均值才有意义。只有用经过映射的每种顺序的频率计算时,比率也才有意义。因此,比率和平均值可用来代表相同顺序频率的早期和后来情况之间或两个类似情况之间的差。否则,它们可以用来相互比较每种顺序的频率。

例如:学校的考试成绩:优、良、及格和不及格。

含义说明:每个量取决于它们在顺序中的位置,如中值。

## c) 间隔标度

$M' = aM + b$  ( $a > 0$ )

当两次测度之间的差值有经验意义时,间隔标度包括排序的等级标度。但间隔标度中两次测度的比率可能没有相同的经验意义。

例如:温度(摄氏、华氏、开氏),实际计算时间与预测的时间的差。

含义说明:算术平均值和任何依赖排序的值。

## d) 比率标度

$M' = aM$  ( $a > 0$ )

当两次测度之间的差值及两次测度的比例有相同的经验意义时,比率标度包括排序的等级标度。平均值和比率有各自的含义,它们给出了值的实际含义。

例如:长度、重量、时间、规模、计数。

含义说明:几何平均、百分比。

## e) 绝对标度

$M' = M$ ,它们只能按一种方式测量。

任何与测度有关的说明都是有意义的。例如,当测量的单位相同时,一个比率标度类型的测度除以一个比率标度类型的测度,结果是一个绝对值。一个绝对标度类型的测量值事实上不带任何单位。

例如,带注释的代码行数除以代码的总行数。

含义描述:一切事情。

## C.2 测度类型

### C.2.0 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,度量的用户宜标识并考虑度量所使用的测度类型。

#### C.2.1 规模测度类型

##### C.2.1.0 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。

##### C.2.1.1 功能规模类型

功能规模是软件可能有的的一种规模类型(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:

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

GB/T 18491.1—2001 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,必须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,也不能保证它们依赖于 GB/T 18491.1—2001 标准。不过,在软件开发中,如下的方法仍被广泛使用。

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

##### C.2.1.2 程序规模类型

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。

###### 1. 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

开发者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

开发者可能使用变更的程序规模来代表包含修改过的部件的软件规模;

注3:计算的程序规模

计算程序规模的公式的例子:新代码行 + 0.2 \* 修改过的部件中的代码行(NASA Goddard)

可能有必要更详细地区分下列源代码语句的类型:

###### 1. 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。

## II. 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;

## III. 源

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)

## 2. 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:

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

## 3. 模块数

本测量计算可独立执行的对象个数,例如程序中的模块个数。

### C.2.1.3 利用的资源规模测度类型

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 的利用率或在并发/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,例如文档的页数。

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。

### C.2.1.4 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。

## C.2.2 时间测度类型

### C.2.2.0 导引

时间测度类型度量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) 部件的时间单位

在有多个站点时,部件时间单位标识各个站点,部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部件的可靠性,如部件的失效率。

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。

### C.2.2.1 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。

### C.2.2.2 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。

### C.2.2.3 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)。

### C.2.2.4 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

开发者、维护者或操作者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(开发者、维护者、操作者、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。

### C.2.2.5 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。

### C.2.3 计数测度类型

#### C.2.3.0 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。

#### C.2.3.1 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。

#### C.2.3.2 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。

#### C.2.3.3 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或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者、客户、操作者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。

#### C.2.3.4 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。

#### C.2.3.5 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,可按严重性的等级进行分类。

#### C.2.3.6 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。

#### C.2.3.7 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。

#### C.2.3.8 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。



附 录 D  
(资料性附录)  
术 语

D.1 定义

除非特别指出,全部的定义都引自 GB/T 18905.1—2002 和 GB/T 16260.1—2006。

D.1.1 质量 quality

外部质量 external quality

产品在特定条件下使用时,满足明确或隐含要求的程度。

内部质量 internal quality

产品属性的总和,决定了产品在特定条件下使用时,满足明确和隐含要求的能力。

注 1:当术语“特性”在本标准中用于更特定的意思时,使用术语“属性”(而不是 3.1.3 中使用的术语“特性”)。

质量 quality

实体特性的总和,表示实体满足明确或隐含要求的能力。

注 2:在某种契约的环境或在某个受控的环境中,如核安全领域,要求是明确规定的,而在其他环境中,宜确定和定义隐含的要求。

使用质量 quality in use

特定用户使用产品满足其要求的程度,以达到在特定应用环境中的有效性、生产率和满意度等特定目标。

注 3:使用质量是在包含软件的环境中质量的用户观点,它可以用在环境中使用软件的结果来测量,而不是根据软件本身的性质来测量。

注 4:在 GB/T 18905.1—2002 中使用质量的定义目前不包括新的“安全性”特性。

质量模型 quality model

一组特性及特性之间的关系,它提供规定质量需求和评价质量的基础。

D.1.2 软件与用户 software and user

软件 software

信息处理系统的部分或全部程序、过程、规则及相关的文档。[GB/T 5271.1—1993]。

注 1:软件是独立于所记录媒体的智力创作。

软件产品 software product

一组计算机程序、规程以及可能有的相关文档和数据。[GB/T 8566—2001]

注 2:产品包括中间产品及专为开发者或维护者这样的用户所准备的产品。

用户 user

使用软件产品执行特定功能的个人。

注 3:用户可以包括操作者、软件结果的接受者或软件的开发者或维护者。

D.1.3 测量 measurement

属性 attribute

实体的可以测量的物理或理论上的性质。

直接测度 direct measure

不依赖于任何其他属性测度的一种对属性的测度。

外部测度 external measure

从对系统行为的测度导出的对产品的一种间接测度,其中产品是系统的一部分。

注 1:系统包括任何相关的硬件、软件(定制的软件或现货软件)和用户。

注 2：在测试中发现的故障数量是对程序中的故障数量的外部测度，因为故障的数量是在计算机系统运行程序的过  
程中计数，以便标识代码中故障的数量。

注 3：外部测度可以用来评价更接近于最终设计目标的质量属性。

#### **指标 indicator**

能用来估计或预测另一测度的一种测度。

注 4：预测的测度可以是针对相同或不同的软件质量特性。

注 5：指标可用来估计软件质量的属性和开发过程的属性，它们是对属性的不精确的间接测度。

#### **间接测度 indirect measure**

从一个或一个以上的其他属性的测度导出的一种对属性的测度。

注 6：对计算机系统属性(例如对用户输入的响应时间)的外部测度就是对软件属性的一种间接测度，因这种测度要  
受计算环境的属性和软件属性的影响。

#### **内部测度 internal measure**

对产品本身的一种测度，或是直接的或是间接的。

注 7：代码行数、复杂度、在走查和 Fog 索引中发现的故障数都是对产品本身进行的内部测度。

#### **测度(名词) measure(noun)**

通过执行一次测量赋予实体属性的数或类别。

#### **测量 measurement**

使用一种度量，把标度值(可以是数或类别)赋予实体的某个属性。

注 8：“类别”可用于指示属性的定性测度。如软件产品的一些重要属性，例如源程序语言(Ada, C, COBOL 等)就是  
定性的类别。

#### **度量 metric**

定义的测量方法和测量标度。

注 9：度量可以是内部的或外部的，直接的或间接的。

度量包括把定性数据进行分类的方法。

---