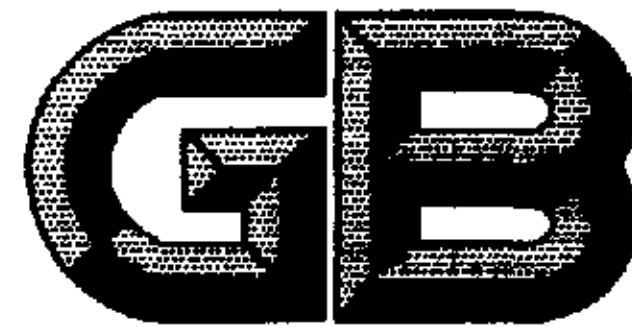


ICS 35.080
L 77



中华人民共和国国家标准

GB/T 18905.1—2002/ISO/IEC 14598-1:1999

软件工程 产品评价 第1部分:概述

Software engineering—
Product evaluation—
Part 1: General overview

(ISO/IEC 14598-1:1999, Information technology—Software product
evaluation—Part 1: General overview, IDT)

2002-12-04 发布

2003-05-01 实施

中华人民共和国
国家质量监督检验检疫总局 发布

前 言

GB/T 18905—2002《软件工程 产品评价》分为六个部分：

- 第1部分：概述；
- 第2部分：策划和管理；
- 第3部分：开发者用的过程；
- 第4部分：需方用的过程；
- 第5部分：评价者用的过程；
- 第6部分：评价模块的文档编制。

本部分为 GB/T 18905—2002 的第1部分，等同采用 ISO/IEC 14598-1:1999《信息技术 软件产品评价 第1部分：概述》(英文版)。

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：北京信息工程学院、中国电子技术标准化研究所。

本部分主要起草人：王凌、冯惠、罗锋盈、陈莹。

引 言

随着信息技术应用的不断增长,关键的计算机系统的数量也在增长。这些系统包括:安全、生活、经济以及保密方面的关键系统。这些系统的软件质量尤其重要,因为软件的故障可能导致非常严重的后果。

纵观整个软件工程的历史,提高软件质量已成为最重要的目标。评价软件产品的质量对获取和开发满足质量需求的软件是不可缺少的。各种软件质量特性的相关重要性取决于作为整体一部分的任务和目标,需要评价软件产品以判断其相关的质量特性是否满足系统的需求。

软件质量评价的基本部分包括:质量模型、评价方法、软件的测量和支持工具。要想开发好的软件,宜规定质量需求,宜策划、实现和控制软件质量保证过程,宜评价中间产品和最终产品。要达到评价软件质量的目的,宜用有效的度量方法进行测量软件的质量属性。

术语“度量”在软件工程出版物中有多种含义,在本部分中被定义为用于测量的一种量化的标度和方法。名词“度量”用来指测量的结果。

GB/T 18905 系列标准为软件产品质量的测量、评估和评价提供了方法。它所描述的既不是软件生产过程的评价方法,也不是预算成本的方法(软件产品的质量测量当然可以用于这两个目的)。

软件工程 产品评价 第 1 部分:概述

1 范围

GB/T 18905 的本部分介绍了 GB/T 18905 的其他部分。它给出了其他部分的概述,解释了 GB/T 18905 与 ISO/IEC 9126 标准中质量模型的关系。本部分还定义了其他部分用到的技术术语,包含了软件质量规格说明和评价的一般需求,阐明了一般概念。此外,它还提供了一个评价各种类型的软件产品质量的框架,说明了软件产品测量与评价方法的需求。

GB/T 18905 旨在提供给软件的开发者、软件的需方和独立的评价者使用,特别是供那些负责软件产品评价的人员使用。应用 GB/T 18905 所产生的评价结果可供管理者、开发者或维护者测量软件是否符合需求,并在必要的地方予以改进。分析人员可用评价结果来建立内部度量与外部度量间的关系。过程改进人员可用评价结果来确定如何通过研究和检查项目的产品质量信息来改进过程。

注: GB/T 18905 中的大多数指南并不特别针对软件,也适用于其他的复杂产品。

2 一致性

如果使用了第 6 章中的过程和 8.3 中要求的质量模型,软件的规格说明和评价就与 GB/T 18905 一致。与 GB/T 18905 整体上一致意味着与 GB/T 18905 所有已公布的部分一致。

3 规范性引用文件

下列文件中的条款通过本部分的引用而成为本部分的条款。凡是注日期的引用文件,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的各方研究是否可使用这些文件的最新版本。凡是不注日期的引用文件,其最新版本适用于本部分。

- GB/T 5271.1—2000 信息技术 词汇 第 1 部分:基本术语(idt ISO/IEC 2382-1:1993)
- GB/T 6583—1994 质量管理和质量保证 词汇(idt ISO 8402:1994)
- GB/T 8566—2001 信息技术 软件生存周期过程(idt ISO/IEC 12207:1995)
- ISO/IEC 9126-1 软件工程 产品质量 第 1 部分:质量模型

4 术语和定义

下列术语和定义适用于 GB/T 18905 的所有部分。

4.1

需方 acquirer

从供方获得或采购系统、软件产品或软件服务的组织。

[GB/T 8566—2001]

4.2

属性 attribute

实体的可以测量的物理或理论上的性质。

注:属性可以是内部的或外部的。

GB/T 18905.1—2002/ISO/IEC 14598-1:1999

4.3

开发者 developer

在软件生存周期过程中执行开发活动(包括需求分析、设计、测试直至验收)的组织。

[GB/T 8566—2001]

4.4

直接量测 direct measure

不依赖于任何其他属性度量的一种对属性的度量。

4.5

评价模块 evaluation module

针对特定软件质量特性或子特性的评价技术包。

注：该评价技术包概括了评价方法和技术、要评价的输入、要测量和收集的数据，以及支持规程和工具。

4.6

外部测量 external measure

通过对系统行为的测量得出的对产品的一种间接量测，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：在测试中发现的失效数量是对程序中的故障数量的外部度量，因为失效的数量是在计算机系统运行程序的过程中计算的。

注3：外部测量可以用来评价更接近于最终设计目标的质量属性。

4.7

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。

4.8

失效 failure

产品完成所需功能的能力的终止，或在原先规定的限制内没有能力完成。

4.9

故障 fault

计算机程序中的不正确的步骤、过程或数据定义。

注：该定义取自 IEEE 610.12—1990。

4.10

隐含的要求 implied needs

当实体用在特定条件下时，尚未说明但又是实际需要的要求。

注：隐含的要求是未形成文档的真实要求。

4.11

指标 indicator

能用来估计或预计另一度量的一种度量。

注1：预计的度量可以有相同或不同的软件质量特性。

注2：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量。

4.12

间接量测 indirect measure

从一个或一个以上的其他属性的测量得出的一种对属性的测量。

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

4.13

软件中间产品 intermediate software product

软件开发过程中的产品,用作软件开发过程另一阶段的输入。

注:在某些情况下,中间产品也可以是最终产品。

4.14

内部度量 internal measure

对产品本身的一种度量,或是直接的或是间接的。

注:代码行数、复杂度度量、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度量。

4.15

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:在 GB/T 18905 中,与术语“外部质量”对应的“内部质量”基本上与 GB/T 6583—1994 中“质量”的含义相同。

注2:术语“属性”与 4.21 中术语“特性”的含义相同,而术语“特性”在 ISO/IEC 9126 中有更具体的含义。

4.16

维护者 maintainer

执行维护活动的组织。

[GB/T 8566—2001]

4.17

量测 measure(verb)

执行一次测量。

4.18

度量 measure(noun)

通过执行一次测量赋予实体属性的数字或类别。

4.19

测量 measurement

使用一种度量,把标度值(可以是数字或类别)赋予实体的某个属性。

注:使用类别时,测量可以是定性的。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。

4.20

度量(体制) metric

定义的测量方法和测量标度。

注1:度量(体制)可以是内部的或外部的,可以是直接的或间接的。

注2:度量(体制)包括把定性数据进行分类的方法。

4.21

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1: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(GB/T 6583—1994,注1)。

注2:在 GB/T 18905 中相关的实体是指软件产品。

[GB/T 6583—1994]

4.22

质量评价 quality evaluation

对实体能满足特定需求的程度的系统检测。

注:当按照合同为某个特定用户开发产品时,其需求是正式规定的;当产品是为非特定用户开发时,如消费软件,其需求由开发组织来规定;当用户为比较和选择的目的评价产品时,需求可以是更一般的。

GB/T 18905.1—2002/ISO/IEC 14598-1:1999

[GB/T 6583—1994]

4.23

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:这种使用质量的定义类似于 ISO 9241-11 中可用性的定义。在 GB/T 18905 中术语“可用性”用来指在 ISO/IEC 9126 中描述的软件质量特征。

4.24

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。

4.25

评级 rating

把测量值映射到相应的评定等级的活动,用于确定与软件某一质量特性相关的等级。

4.26

评定等级 rating level

在有序标尺上的某个刻度,用于分类某一测量的标度。

注1:评定等级能使软件按照明确或隐含的要求进行分类(评定)(见 10.2)。

注2:相应的评定等级与质量的不同角度有关,如“用户”、“管理者”或“开发者”的角度。

4.27

标度 scale

具有特性定义的一组值。

注:标度类型的例子有:与一组类别对应的标称标度、与一组有序刻度对应的序数标度、与一组等距的有序刻度对应的间隔标度,以及既有等距刻度,也具有绝对零度的比例标度。使用标称标度或序数标度的度量(体制)产生定性的数据,而使用间隔标度和比例标度的度量(体制)产生定量的数据。

4.28

软件 software

信息处理系统的部分或全部程序、规程、规则及相关的文档。

注:软件是独立于所记录媒体的智力创作。

[GB/T 5271.1—2000]

4.29

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。

注:产品包括中间产品和打算由开发者和维护者等用户使用的产品。

[GB/T 8566—2001]

4.30

供方 supplier

同需方签订合同,并按合同的规定提供系统、软件产品或软件服务的组织。

[GB/T 8566—2001]

4.31

系统 system

由一个或多个过程、硬件、软件、设施和人员组成的集合体,提供满足明确要求或目标的能力。

[GB/T 8566—2001]

4.32

用户 user

使用软件产品执行特定功能的个人。

注：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。

4. 33

确认 validation

通过检查和提供客观证据证实某一规定预期用途的特殊需求已经满足。

注 1：在设计和开发中，确认关系到检查产品是否符合用户要求的过程；

注 2：确认一般是在规定的操作条件下对最终产品进行的。在早期阶段，这样做是必要的；

注 3：“确认过的”一词用来表示相应的状况；

注 4：如果有几种不同的预期用途，可进行多种确认。

[GB/T 6583—1994]

4. 34

验证 verification

通过检查和提供客观证据证实规定的需求已经满足。

注 1：在设计和开发中，验证是指对某项指定活动的结果进行检查的过程，以确定该活动是否符合明确的需求。

注 2：“验证过的”一词用来表示相应的状况。

[GB/T 6583—1994]

5 GB/T 18905 及 ISO/IEC 9126 的概述

5.1 GB/T 18905 及 ISO/IEC 9126 的结构

ISO/IEC 9126 标准定义了一个通用的质量模型和质量特性，给出了度量的例子。GB/T 18905 概述了软件产品评价的过程，提供了评价需求和指南。GB/T 18905. 2 和 GB/T 18905. 6 与公司或部门级的评价管理和支持有关，GB/T 18905. 3、GB/T 18905. 4 和 GB/T 18905. 5 给出了项目级的评价需求和指南。图 1 和图 2 示出了这些标准和技术报告之间的关系。

5.2 评价过程

GB/T 18905 给出了在 3 种不同情况下评价过程的需求和指南：

- 开发(增强)(GB/T 18905. 3)；
- 获取(GB/T 18905. 4)；
- 独立评价(包括第三方评价)(GB/T 18905. 5)。

5.2.1 开发者用的过程

计划开发新产品或增强现有的产品，以及打算利用他们自己的技术人员进行产品评价的组织宜使用 GB/T 18905. 3。这部分主要强调使用那些能预测最终产品质量的指标，这些指标将通过度量在生存期期间开发的中间产品来得到。

5.2.2 需方用的过程

计划获取或复用某个已有的软件产品或预先开发的软件产品的组织宜使用 GB/T 18905. 4。该部分可用来决定接受产品或者从众多可选产品中选择某个产品。(产品可以是自包含的，或是系统的一部分，或者是较大产品的一部分。)

5.2.3 评价者用的过程

对软件产品执行独立评估的评价者宜使用 GB/T 18905. 5。这种评价是应开发者、需方或其他方的请求来进行的。这部分将由那些执行独立评价的人员使用，他们通常为第三方组织进行工作。

5.3 对评价的支持

上述每个评价过程的标准都能与 GB/T 18905. 2(策划和管理)和 GB/T 18905. 6(评价模块的文档编制)结合起来使用(图 1)。

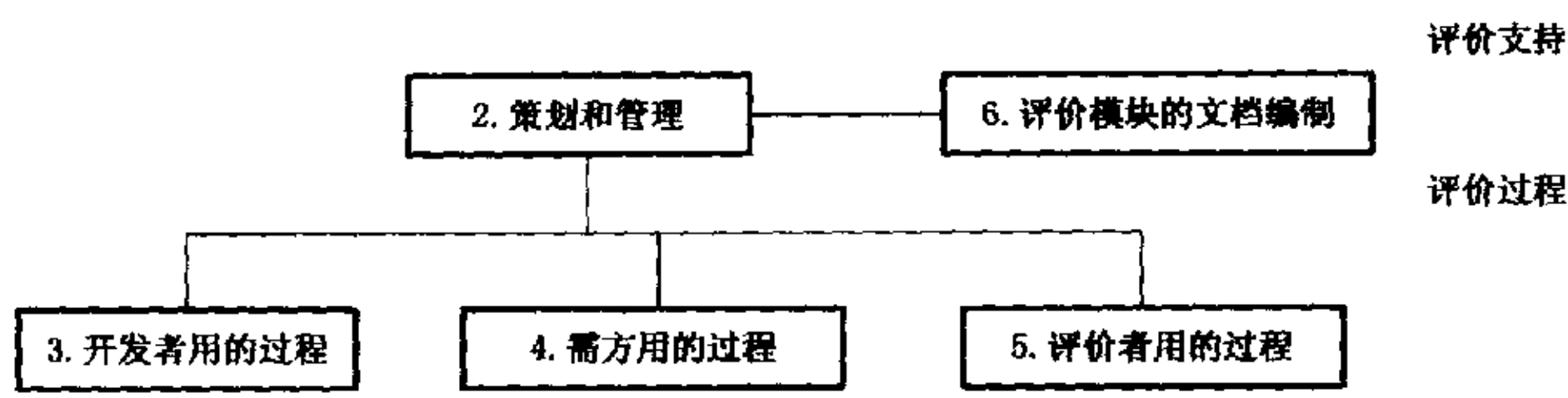


图 1 评价过程与评价支持标准的关系

5. 3. 1 策划和管理

GB/T 18905. 2——策划和管理部分包含对软件产品评价的支持功能的需求和指南。这种支持与策划和管理软件评价过程及相关的活动有关,包括组织内评价专业知识的开发、获取、标准化、控制、转换和反馈。本部分可供管理者制定一个定量的评价计划。

5. 3. 2 评价模块

GB/T 18905. 6 为编制评价模块的文档提供指南。这些模块包括质量模型的规范(即特性、子特性和相应的内部或外部度量)、与模型计划的应用有关的数据和信息和与模型的实际应用有关的信息。每种评价均应选择适当的评价模块。对某些情况,还有必要开发新的评价模块。本部分可供组织用来产生新的评价模块。

5. 4 软件质量特性和度量

GB/T 18905 的各个部分宜与 ISO/IEC 9126 中描述软件质量特性和度量的下列部分一起使用:

- 质量特性和子特性(ISO/IEC 9126-1);
- 外部度量(ISO/IEC 9126-2);
- 内部度量(ISO/IEC 9126-3)。

ISO/IEC 9126-1 定义了质量特性、相关的子特性以及 ISO/IEC 9126 质量模型上面三层之间的关系。ISO/IEC 9126-2 和 ISO/IEC 9126-3 确定了每种度量(外部和内部的)与其相应的特性和子特性之间的关系(见图 2)。注意,某些内部度量有对应的外部度量。

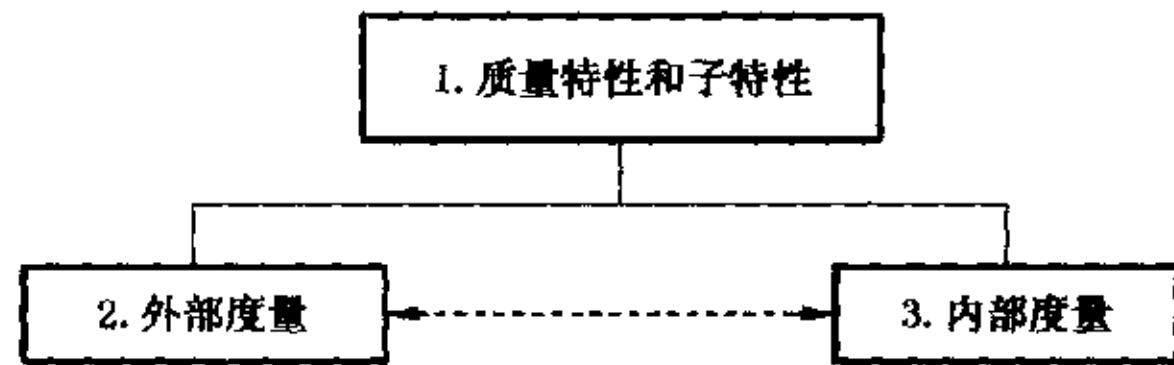


图 2 ISO/IEC 9126 软件质量特性和度量

5. 4. 1 质量特性和子特性

ISO/IEC 9126-1 定义了一组质量特性和相应的子特性。当把软件作为计算机系统的一部分来使用时,这些子特性表现为外部子特性,并作为软件内部属性的一个结果。ISO/IEC 9126-1 用作构造质量模型上面三层的基础。从用户的角度看,质量的总目标为使用质量。

5. 4. 2 外部度量

ISO/IEC 9126-2(在制定中)描述了使用软件时代表软件质量外部特征的那些度量。在使用软件时的某个预定义阶段采用外部度量,如类似时间和工作量等量化值用作这些度量的基础。这些度量应用于测试和运行阶段。当在测试阶段使用时,运行该软件,就意味着这些度量将成为所期望的质量等级的初始预测值。这些度量一般说来代表与用户相关的质量。

5. 4. 3 内部度量

ISO/IEC 9126-3(在制定中)描述了测量有关设计和代码等软件内部属性的那些度量。一旦测试和运行软件系统,这些“初期”的度量就用作预测期望达到的指标。因此,对开发管理者来说,内部度量是最重要的,因为它是预先防范出现连带问题的有效方法。用内部度量来预测相应的外部度量值。ISO/

IEC 9216-3 显示了哪些内部度量有对应的外部度量。

6 评价过程

要评价软件的质量,首先要确立评价需求,然后规定、设计和执行评价(见图 3)。每个步骤都在指定的章节中有更详细的描述。本部分给出了对整个过程的概述。GB/T 18905 的其他部分解释了如何在不同情况下应用该过程。

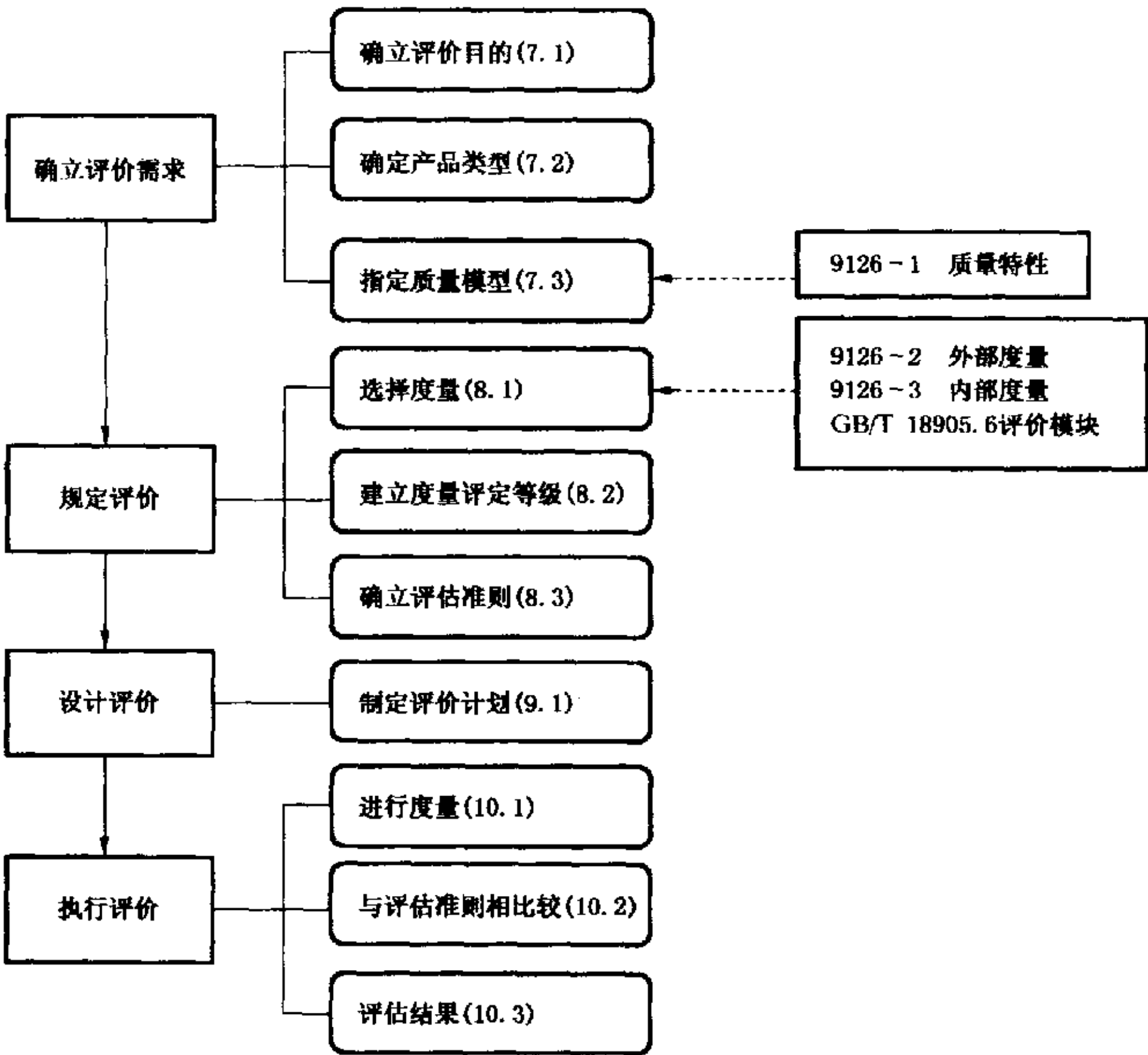


图 3 评价过程

7 确立评价需求

7.1 确立评价目的

7.1.1 概述

软件质量评价的目的是为了直接支持开发和获得能满足用户和消费者要求的软件。最终目标是保证产品能提供所要求的质量,即满足用户(包括操作者、软件结果的接受者,或软件的维护者)明确和隐含的要求。

评价中间产品质量的目的是:

- 决定(是否)接受分包商交付的中间产品;
- 决定某个过程的完成,以及何时把产品送交下一个过程;
- 预计或估计最终产品的质量;
- 收集中间产品的信息以便控制和管理过程。

评价最终产品质量的目的是:

- 决定(是否)接受产品;
- 决定何时发布产品;

GB/T 18905.1—2002/ISO/IEC 14598-1:1999

- 与相互竞争的产品进行比较；
- 从众多可选的产品中选择一种产品；
- 使用产品时评估产品的积极和消极的影响；
- 决定何时增强或替换该产品。

评价软件的质量可以在一个已经定义的质量结构中进行,这种质量结构将涉及在 GB/T 8566 中定义的产品开发和获取生存周期的全过程。

7.1.2 获取

当获取某个客户制造的软件产品时,宜由需方确立外部质量需求,规定对供方的需求,并在获得产品之前对照这些需求来评价潜在的购买。

开发某个产品时,规定质量需求的目的是为了确保该产品符合用户的明确和隐含的要求(见 GB/T 18905.3)。

购买某个软件产品时,可以通过评价来比较几种替代的产品,并确保所选的产品能满足质量需求(见 GB/T 18905.4;需方用的过程,GB/T 18905.5;评价者用的过程,包括第三方的评价需求)。

7.1.3 供应

供方可以利用软件产品的评价结果来保证产品满足所需的质量准则,这一准则由需方来确定,或与其他产品进行比较来确定。

7.1.4 开发

软件产品的需求表达了用户考虑的对软件产品的要求,这些要求在开发之前定义。当软件产品被分解成几个主要部件时,对整个产品的需求相对于不同的部件可能是不同的,需要不同的评价准则。在评价质量之前,宜根据质量特性和子特性规定质量需求(见 ISO/IEC 9126-1)。

在评价的最初阶段,宜研究和确定针对策划评价和实施评价的质量需求。开发者宜对每个相关的质量特性确立外部质量需求。宜评价这些质量需求规格说明的完整性和正确性,以确保所有必要的需求都已规定,所有不必要的需求都被排除。开发者需要在产品交付之前根据这些需求评价其产品。

为了达到必须满足的明确和隐含的质量需求,重要的是要检查所有相关质量特性的隐含要求是否规定得非常详细。如果有可能,宜由中介方或购买方来评估需求,由最终用户来评估隐含的要求。用户对原型产品的使用经验会有利于更准确地描述使用质量需求。

宜由开发者来确定内部质量需求。使用内部质量需求时,开发者宜采用将内部质量需求与外部质量需求相对应的质量模型来确定这些内部质量需求,并在开发过程中使用内部需求来验证中间产品的质量。

宜通过对开发过程中的中间产品规定内部质量需求的方式,使软件评价能用来预计和验证开发过程中的质量。而针对具体用途的完整产品的外部质量则可以根据最初的需求进行评价(见 GB/T 18905.3,开发者用的评价过程)。

软件质量评价的结果可被用来获得使用不同的开发过程、不同的设计方法或不同的 CASE 工具,从而满足不同程度的质量需求的反馈。

7.1.5 运行

运行软件系统的组织可以通过软件质量的评价来确认在不同操作条件下满足质量需求的情况,并且对那些负责维护的人员提供对修改要求的反馈。

7.1.6 维护

维护软件系统的组织可以通过软件的评价来确认是否仍然满足质量需求,以及确认是否达到可维护性和可移植性的需求。

7.2 确定要评价产品的类型

要评价的中间或最终软件产品的类型取决于所处的生存周期的阶段和评价的目的(见图 4)。

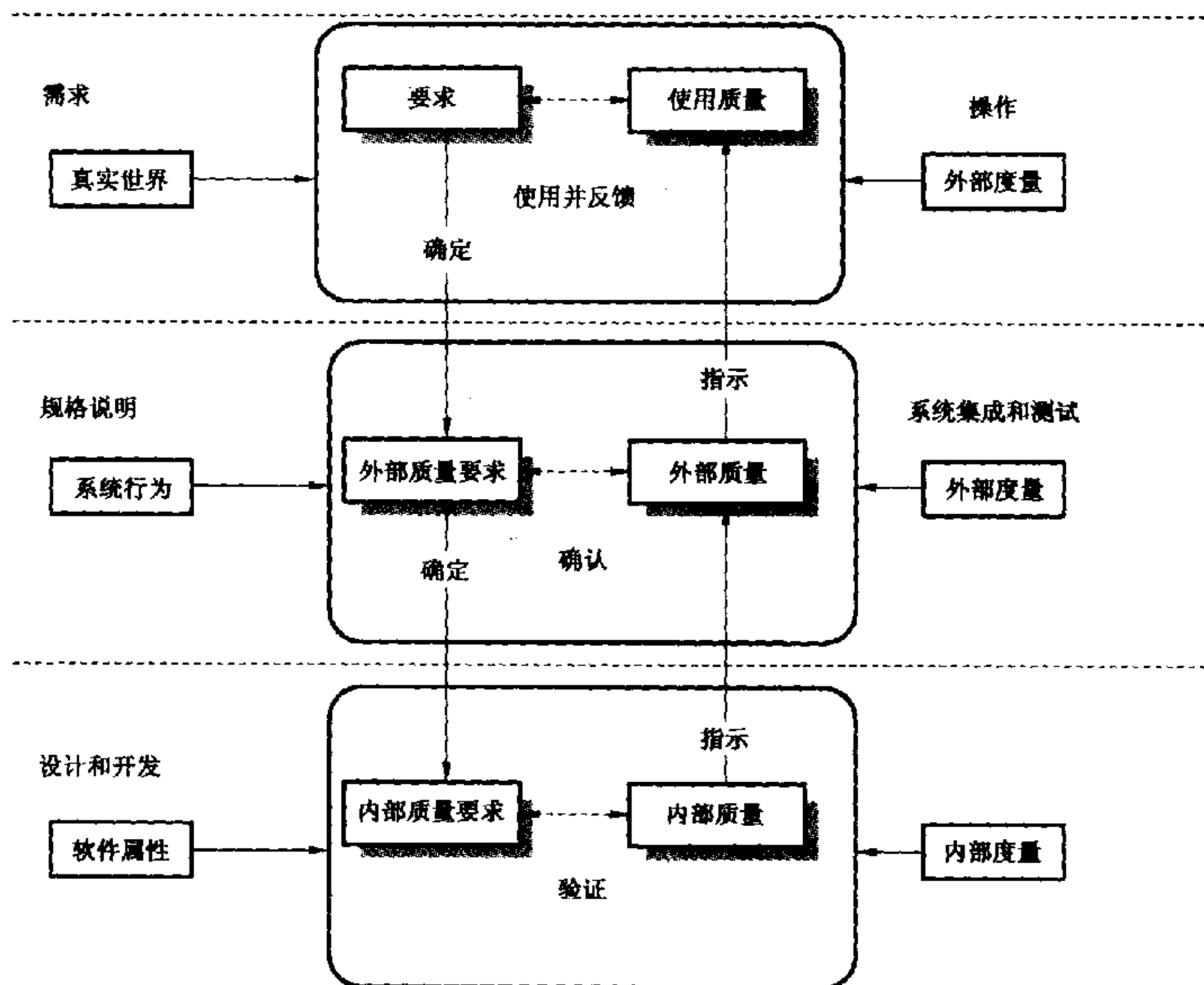


图4 软件生存周期中的质量

其目标是当用户实际使用该软件产品时，它能满足明确和隐含的要求。只能针对包含该软件产品的整个硬件或软件系统评估其外部质量。执行该软件时应用外部度量。由于外部度量值不只是依赖于软件，所以软件必须作为某个工作系统的一部分来评价。

使用质量是针对特定用户(可以是最终用户、操作者或维护者)的关于质量特性的综合效应。具有使用质量的软件必须能满足用户的要求，以便能在特殊的硬件和软件环境中执行特定的任务。在某种环境中能顺利执行任务的软件，在另一种环境中可能会出故障。因此对质量特性的外部评价宜在尽可能准确地模拟期望使用的那些条件下进行，即使不可能模拟真实的使用条件(如网络环境和用户特性)，也要在编码完成时对特性进行外部度量，因为外部度量经常只作为实际使用质量的指标。

如果未达到外部质量的需求，则评价结果可被用作修改软件特性的反馈，以便提高外部质量，从而支持一个循环反复的改进过程。

为进一步开发，要定义能使中间产品质量得到确认的内部质量需求。软件的内部特性(如规格说明或源代码)可利用内部度量来测量。在开发过程中，最有意义的是内部度量。内部度量能用作外部属性的指标。模块化和可追溯性是两个能被测量的内部属性的例子。达到所要求的内部质量会有助于满足应用软件的外部需求。因此，可以把对软件内部质量的度量用作评估使用质量的指标(见图5)。

例如，响应时间是评价软件易用性和效率的一项重要度量，但响应时间不能在开发期间测量。为了在开发过程中评价产品效率，可根据中间产品或规格说明来测量路径的长度，该长度可用作在某种条件下粗略计算响应时间的一个指标。

将软件的内部质量属性与外部质量需求相关联是十分重要的，使得开发中的软件产品(中间和最终的软件产品)的质量特性可以根据最终系统使用质量需求来进行评估。内部度量的值很低，除非有证据表明它与外部质量有关。

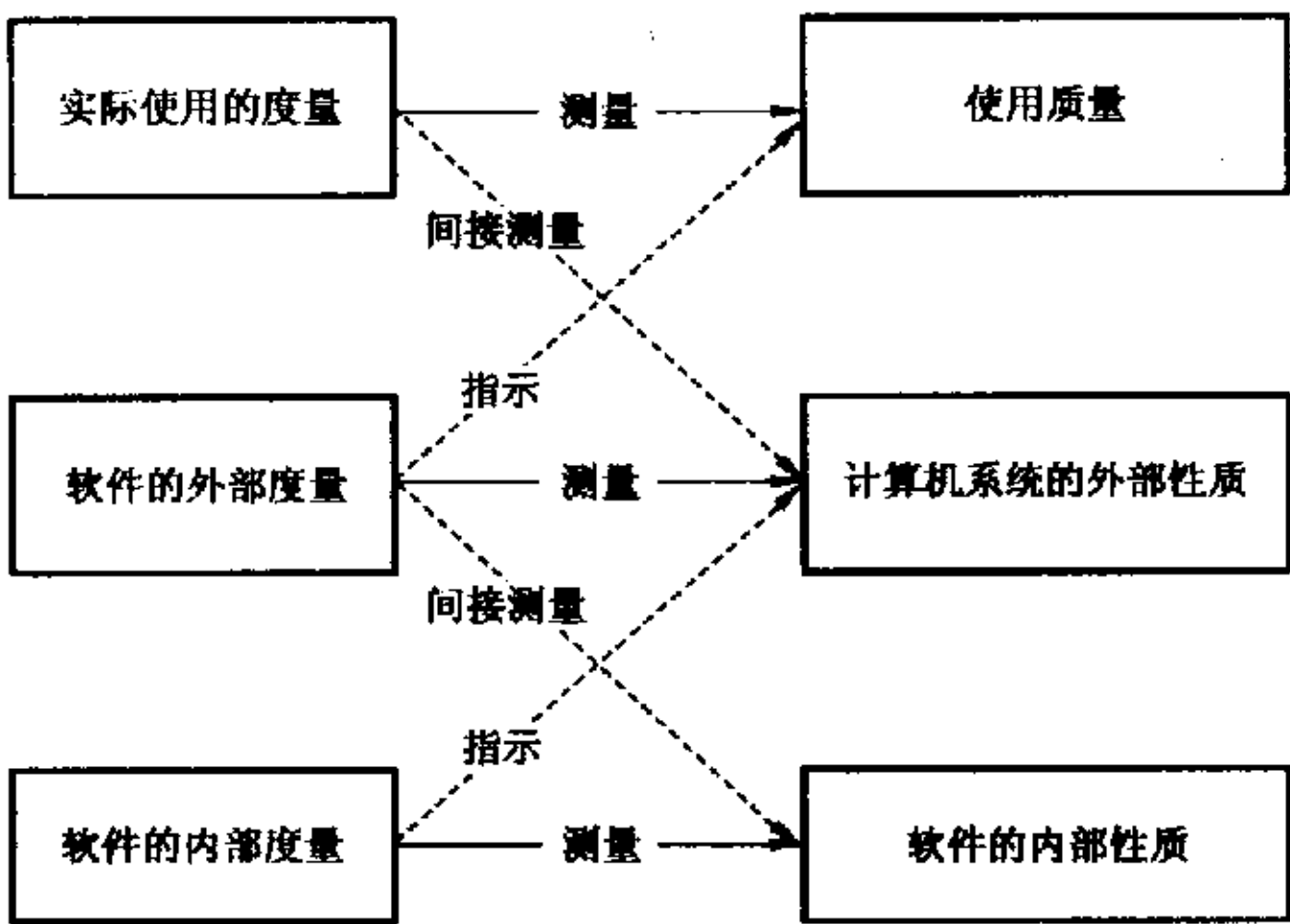


图 5 度量之间的关系

与最终质量有关的特定属性取决于所要求的使用条件。对于交互产品，这将取决于实际最终用户和任务的需要。其他影响软件产品质量需求的方面包括：购买产品还是开发产品，产品的开发阶段，产品要用到的硬件、软件和网络环境等。

对计算机系统的外部度量也可用作对软件内部质量的间接测量。因此，计算机系统的响应时间可被用来测量软件在特定计算环境中的效率。

7.3 规定质量模型

软件评价的第一步是选择相关的质量特性，使用一个将软件质量分解成几种不同特性的质量模型。软件评价所用的质量模型通常代表软件质量属性的总体，这些质量属性用特性和子特性的分层树结构进行分类。该结构的最高级由质量特性构成，最低级由软件质量属性构成。ISO/IEC 9216-1 提供了一个通用模型，它定义了 6 种软件质量特性，包括：功能性、可靠性、易用性、效率、可维护性和可移植性。这些特性还能进一步被分解为具有可测量属性的子特性。在特定的使用环境下，质量特性的组合效应被定义为使用质量。

软件产品的内部质量属性是软件产品的可测量性质，这些性质影响产品的满足明确和隐含要求的能力。可以用一个或更多的属性来评估一个特定的软件质量特性或子特性（见图 6）。

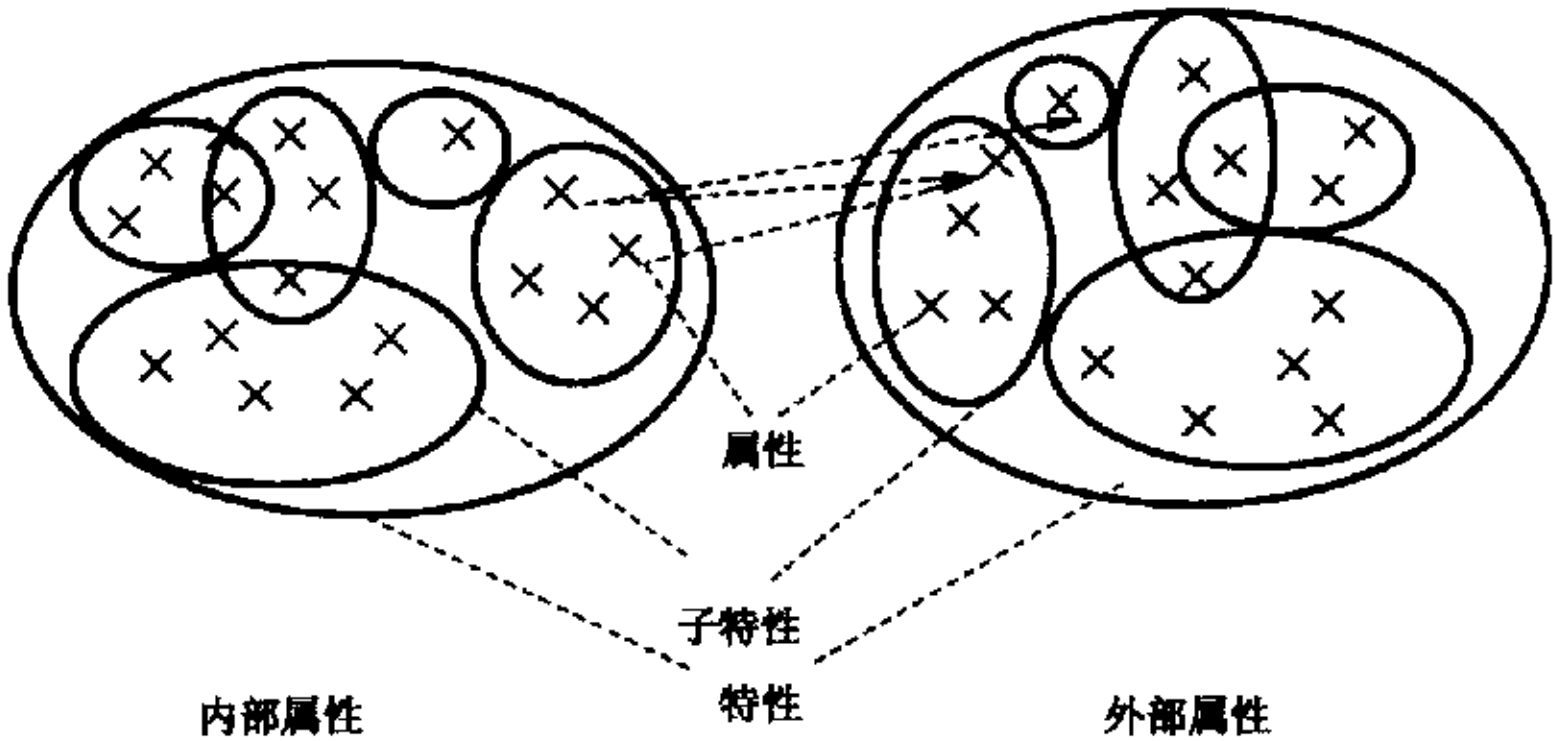


图 6 质量特性、子特性和属性

宜对每个要求的子特性确定其足够的内部属性和外部属性。

与任何特殊情况有关的实际特性和子特性将取决于评价的目的，宜通过对质量需求的研究来确定。ISO/IEC 9216-1 中的特性和子特性提供了一个有用的、关于质量的问题检查表，但是质量分类的其他方法可能更适用于某些特定的环境。

注：一个例子是 IEC 60050(191)，它把可信性定义为用户可以合理地依靠系统所提供服务的依赖程度。这种程度又分为可靠性、可用性和可维护性等特性。还可以包括易用性、可恢复性、安全性、可扩展性和保密性。

8 规定评价

8.1 选择度量

重要的是软件产品的测量要能既简单又经济地进行,而且测量结果要易于使用。对许多软件的测量可以方便地用某种工具来进行,也可以打包成一个评价模块(GB/T 18905.6)。

质量特性的定义方式不允许对它们进行直接测量。需要建立与软件产品特性相关的度量。与某个质量特性相关的每个可量化的软件内部属性和每个可量化的软件外部属性与其软件环境进行相互作用能被确立为一种度量。

度量可以随环境和应用度量的开发过程阶段的不同而有所区别。用在开发过程的度量宜与用户观点的度量有关,因为从用户视角出发的度量是至关重要的。

注:度量的例子以及如何为每个子特性选择和概括度量的解释说明见 ISO/IEC 9126 的第 2 和第 3 部分(在制定中)。

8.1.1 测量的种类

评价有两个主要目的:

- 确定问题以便解决问题;
- 与可替换的产品进行比较,或对照需求比较产品质量(可以包括认证)。

所需的测量种类取决于评价的目的。如果主要目的是为了了解和纠正缺陷,可以对软件采取多种测量,以便监视和控制改进。为此而进行广泛的测量是有用的,比如检查表和专家意见等。主要需求是正确地测量以确定软件的任何改变对质量的影响。

需要更严格的度量以便进行可以信赖的比较,这种比较可以在产品间进行,也可以与标准值进行比较。测量规程宜测量软件质量的特性(或子特性),规程要求用有效精度进行测量,以便允许设定准则并进行比较。评价规格说明为每种度量规定一个精确的质量模型、测量方法、测量标度和评定等级是很重要的。当用不同的属性进行产品比较时,出自检查表和专家意见的数据可能不太可靠。应允许存在由测量工具或人工差错可能引起的测量误差。

8.1.2 测量的需求

内部度量宜具有可预见的有效性,即它们宜与一些需求的外部准则有关。例如,某个特定软件属性的内部度量宜与软件使用时质量的某些可测量方面有关。重要的是这些测量应分配与常规期望(值)一致的数值:例如,如果测量建议产品是高品质的,那么这一测量宜与满足特定用户要求的产品一致。

有关测量需求的更多信息可参见 ISO/IEC 9126-1。

8.2 确立度量评定等级

可量化的特征可以用度量质量的方法进行定量的测量。其结果是,将测量值映射到某一标度上。这个值本身并不表示满意的等级,因此,这一标度必须根据需求的不同满意度级别分成不同的范围。例如:

- 将标度分成两类:满意和不满意;
- 将标度分成四类:即针对已有产品或可替换产品的当前级、最差级和计划级。定义当前级是为控制新系统不因当前状况而恶化。计划级是指一旦资源可利用,产品即可获得。最差级是指万一产品不符合计划级时用户的可接受边界(图 7)。

8.3 确立评估准则

软件质量需求规格说明应使用定义良好的适当的质量模型来表示。为此,宜使用 ISO/IEC 9126-1 中的质量模型和定义,除非有特殊原因需使用其他模型。

为了评估产品质量,需要总结针对不同特性的评价结果。评价者宜为此准备一个规程,其中对不同的质量特性使用不同的评价准则,每个质量特性又以数个子特性或子特性的加权组合来说明。规程通常还包括如时间和成本等有助于在特定环境下评估软件产品质量的其他方面。

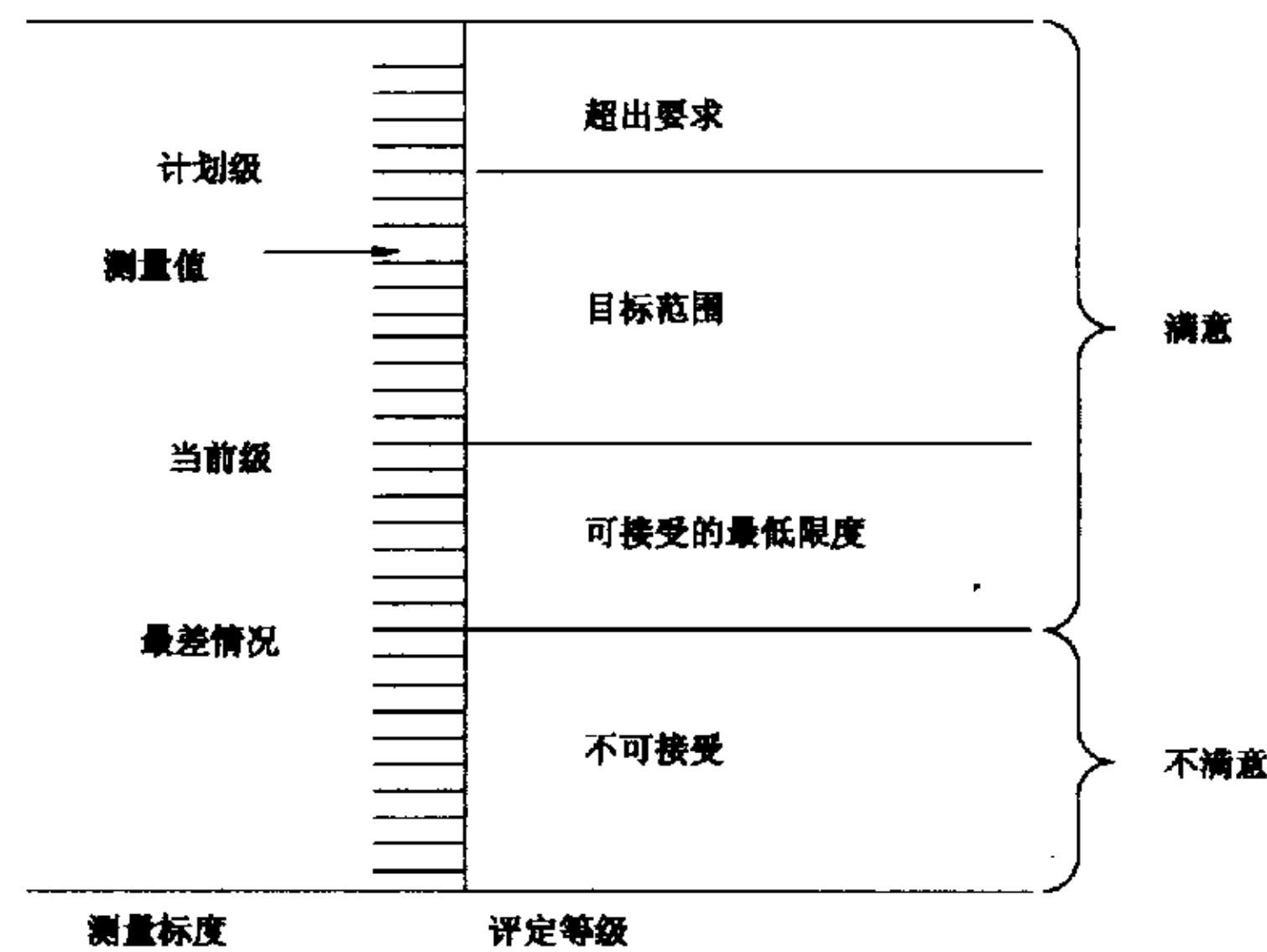


图 7 度量的等级

9 设计评价

9.1 制定评价计划

评价计划描述了评价方法和评价者活动的进度表(见 GB/T 18905. 3、GB/T 18905. 4 或 GB/T 18905. 5)。评价计划宜与测量计划(GB/T 18905. 2)相一致。

10 执行评价

10.1 实施测量

为进行测量,应对软件产品使用所选择的度量。其结果为度量标尺上的值。

10.2 与准则相比较

在评级步骤中,测量的值要与预定的准则(如图 7 所示)进行比较。

10.3 评估结果

评估是软件评价过程的最后一步,将对一组已评定的等级进行概括。其结果是对软件产品满足质量需求程度的一个综述。然后将总结的质量与时间和成本等其他方面进行比较。最后,根据管理准则做出一个管理决策。结果是决策层做出的接受或拒绝、发布或不发布该软件产品的决定。

评价结果对软件开发生存周期的下一步决定十分重要。例如,需求是否须改动? 或开发过程是否需要更多的资源?

11 支持过程

支持评价的活动包括:收集有关评价工具和方法的信息、开发和确认度量、标准化评价过程、度量和测量。GB/T 18905. 2 包含了软件产品评价的支持过程的需求和指南等。

参 考 文 献

国际标准

- [1] IEC 60050-191:1990 国际电子技术词汇 服务的依赖性和质量
- [2] IEEE 610.12—1990 软件工程术语标准
- [3] ISO/IEC 9126-2 软件工程 产品质量 第2部分:外部度量
- [4] ISO/IEC 9126-3 软件工程 产品质量 第3部分:内部度量
- [5] ISO 9241-11:1998 办公用视觉显示终端(VDTs)的人类工效学要求 第11部分:可用性指南

其他参考书目

- [6] Arthur, J. D. and Nance, R. E. , Developing an automated procedure for evaluating software development methodologies and associated products-A final report, Technical report SRC-87-007, Systems Research Center and Virginia Tech, 1987.
- [7] Azuma, M(Editor), Software Quality Evaluation Guide Book(Japanese), JISA, 1994.
- [8] Azuma, M. Software Products Evaluation System: Quality Models, Metrics and Processes-International Standards and Japanese Practice, Information and Software Technology, Vol. No. , 1996, Elsevier.
- [9] Basili, V. R. and Rombach, H. D. , The TAME Project: Towards improvement oriented software environments, IEEE Trans. Software engineering, June 1998.
- [10] Bevan N Measuring usability as quality in use. Journal of Software Quality, 4, p. 115-30, 1995.
- [11] Bevan N and Azuma M(1997) Quality in use: Incorporating human factors into the software engineering lifecycle. In: Proceedings of the Third IEEE international Software Engineering Standards Symposium and Forum(ISESS'97), p. 169-179.
- [12] Boehm, B. w. , et al. , Quantitative evaluation of software quality, Proc. ICSE 1976.
- [13] Bush, M and Russel, M. , Software Engineering Measurement: A Modular Course, Proc. COMP-SAC'91 P571-576.
- [14] Dumke, R. Softwareentwicklung nach Masz(German), Vieweg, 1992.
- [15] Fenton NE and Pfleeger SL, Software Metrics: A Rigorous and Practical Approach, International Thomson Computer Press, 1996.
- [16] McCabe, T. j. A Software Complexity Measure, IEEE Transaction on Software Engineering, Vol. 2, No. 6, 1976.
- [17] McCall et al. , Rome Air Develop Center Report, TR-77-369, 1977.
- [18] Moller, K. H. and Paulish, D. J. Software Metrics, Chapman & Hall(IEEE Press), 1993.
- [19] Miyoshi, Togashi and Azuma, Evaluating software development environment quality, Proc. COMPSAC'89, 1989.
- [20] Natale, D. Qualita E Quantita Nei Sistemi Software(Italian), FrancoAngeli, 1995.
- [21] Sunazuka, T. , and Azuma, M. , Software quality assessment technology, Proc. ICSE 1985.
- [22] Wallmuller, E. Software Quality Assurance A practical approach, Prentice Hall, 1994.
- [23] Zuse, H. , Software Complexity—Measures and Methods, Walter de Gruyter, 1991.