

一种基于关键非功能需求的软件可信性度量模型^{*}

罗新星, 赵玉洁

(中南大学商学院, 长沙 410083)

摘要: 为了更为合理准确地度量软件可信性, 提出了一种基于关键非功能需求的软件可信性度量模型。首先, 分析了该模型所涉及的软件可信性、非功能需求、关键非功能需求的定义; 其次, 对非功能需求进行分解以及权重的分配, 并将非功能需求细分为 A、B、C、D 四类以便更好地获取; 最后, 应用 Web 软件中的“在线旅行”网站详细阐述了该模型的具体度量过程。提出的模型是一个通用的软件可信性度量模型, 适用于不同形态的软件, 支持软件可信性的定制, 对于软件可信性的度量具有一定的指导意义。

关键词: 软件可信性; 非功能需求; 关键非功能需求; 软件可信性度量

中图分类号: TP311 **文献标志码:** A **文章编号:** 1001-3695(2014)04-1077-05

doi: 10.3969/j.issn.1001-3695.2014.04.029

Software trustworthiness measuring model using critical non-functional requirements

LUO Xin-xing, ZHAO Yu-jie

(Dept. of Business, Center South University, Changsha 410083, China)

Abstract: This paper proposed a model using critical non-functional requirements to measure software trustworthiness, and analyzed the definitions including in this model, such as software trustworthiness, non-functional requirements, critical non-functional requirements. Furthermore, it analyzed non-functional requirements decomposition and weights, subdivided and non-functional requirements into A, B, C, D four types for better acquisition. At last, it elaborated a “online travel” website in the Web application software the model of measuring process in detail. The proposed model is a general measurement model of software trustworthiness, applying to different form of software, supporting customizing of software trustworthiness, having certain guiding significance for measuring software trustworthiness.

Key words: software trustworthiness; non-functional requirements; critical non-functional requirements; measurement of software trustworthiness

近十年来,互联网技术的发展催生了信息技术的革命,也使得软件的应用领域极度扩展,社会对软件的需求量以极高的速度增长,对软件的要求以及软件的开发环境也在不断地发生变化。在需求方面,社会需要软件实现的功能越来越复杂,越来越多,并且要求软件具备安全性、可靠性、时效性等特点;在软件的运行和开发环境方面,已经从传统的静态封闭的环境拓展为互联网环境下动态多变的环境。在互联网环境下,计算实体的行为也存在着不可控性和不确定性,这不仅对传统的软件开发方法和技术,也对运行软件的可信性提出了严峻的挑战。

在开放的互联网环境中,构造可信的软件系统和提供可信的运行保障,确保软件为人们提供可信赖的服务、提高软件的可信度是国际软件产业和学术界关注的焦点,更是未来软件技术发展和信息产业发展的关键技术基础。

1 软件可信性

可信软件已经被广泛地应用于航空航天、原子核反应堆、军事科学、国防建设以及金融、证券、医疗卫生、汽车、电力等国民经济的各个高科技领域,它已经直接牵涉到人们的生活质量,甚至关系到人类生命、财产的安全问题^[1]。

在互联网的发展历史中,有许多复杂昂贵的、“成功的”

(也就是已经付款并交付给用户)软件系统由于没有对可信性进行足够的重视而导致失败,这样的例子更是举不胜举(表1)。1995年,美国 Standish Group 著名的 Choas 报告对美国 8 380 个软件项目进行统计,其中仅有 16.2% 的项目按时按预算成功完成,还有 52.7% 的软件项目有问题而超时超预算,其余 31.1% 的项目因完全被取消。

表1 软件可信性事故大事记

时间	内容
1996年6月4日	欧洲 Ariane 5 运载火箭因惯性制导系统中数据转换异常而首发失败
2003年8月14日	美国电力监测与控制管理系统的失效造成美国和加拿大出现当时历史上最大的停电事故,损失超过 60 亿美元
2005年11月1日	日本东京证券交易所因软件系统升级出现故障,导致早间股市“停摆”5 小时
2006年11月2日	美国“火星环球勘测者”飞船由于软件故障而失踪
2006年11月11日	中国中航信离港软件发生 3 次故障,近百个机场值机系统瘫痪
2007年10月30日	北京奥运门票预售首日因系统的实际访问量超过其设计容量使该系统一经启用便陷入瘫痪导致被迫停运 5 日
2009年3月17日	英格兰卡迪夫公司控制吊运构件的计算机系统出现故障,导致 600 公斤重的构件砸在了雇员头上,雇员当场死亡
2012年9月26日	中国 12306 铁路客户服务中心网站日点击量达到 14.9 亿次,导致出现网络拥堵、重复排队等现象

1.1 可信性的认识

“可信”一词本属于社会学的范畴,用来表示主体和主体

收稿日期: 2013-06-13; **修回日期:** 2013-07-29 **基金项目:** 国家自然科学基金委员会创新研究群体资助项目(70921001/G0104)

作者简介: 罗新星(1956-),男,湖南邵阳人,教授,博导,主要研究方向为管理信息系统(star@mail.csu.edu.cn);赵玉洁(1986-),女,甘肃兰州人,博士研究生,主要研究方向为可信软件。

之间或主体与客体之间信任关系的状态。当可信用于主体和主体之间时,具有双向性,但并不保证具有一致性;当用于主体与客体时,因为客体并不具有主观意识,所以可信只具有主体到客体的单向性,客体是无法信任主体的(图1)。

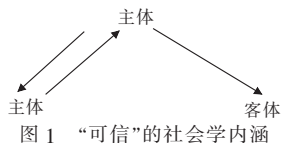


图1 “可信”的社会学内涵

1.2 软件可信性的内涵

人们最初把“可信”一词引入计算机领域时使用的是“dependability”,当时对应研究的是软件的可靠性问题,此时信任的价值判断方向仍是从主体到客体的单方向。当计算机网络技术诞生后,计算机使用的环境发生了巨大的变化,人们在使用计算机时面临了更多更复杂的问题,并且借助于网络,这些问题的影响范围以及问题的严重性都得到了空前的扩大,“可信”的价值判断也不再仅仅存在于单方向的主体到客体,也存在于客体对主体的方向(例如,软件对用户身份的验证)和客体与客体的相互之间(例如,可信计算平台技术要求模块之间的互信验证)。此时人们谈论软件领域的可信问题时,就开始使用“trustworthiness”。可见,从可信的价值判断方向上,trustworthiness 包含了 dependability,从内容上亦是如此。因此,trustworthiness 更加适合描述当前人们所面临的软件可信性问题。

软件可信性内涵的特别之处在于,软件可信性在本质上是传统软件质量概念在互联网时代的延伸,但与传统的软件质量研究的关注点有所不同,它更关注的是使用层面的综合化的质量属性及其保障,由于软件系统越来越复杂,软件可信性意味着软件行为可信、环境可信和使用可信等不同层次的可信性要求,如何从整体上度量、获得并保证软件可信性是非常困难的^[2]。

1.3 软件可信性的定义

随着软件规模越来越大,故障和漏洞在所难免,再加之软件开发和运行的环境已经从封闭、静态到开放、动态转变,导致很多不确定因素存在。软件很多时候并不以人们期望的方式工作,经常发生失效,直接或者间接地给用户带来损失。软件并不总是完全可以让人信任的,其行为和结果并不完全符合人们的预期,这就是所谓的软件可信性问题。

定义1 软件可信性 (software trustworthiness, ST) 是指软件的行为和结果符合用户预期,并在受到干扰时仍能提供连续服务的能力^[3]。

软件可信性是对软件行为的整体刻画,是对软件可靠性、安全性、正确性、时效性等一系列特征属性的综合度量,是软件可信基础研究的核心问题之一。软件可信性度量的问题其实就是软件可信性的属性选取和度量、用户预期的表达和依据对被选择属性的度量值和用户预期对两者符合性的量化问题^[4]。

2 软件可信性度量——非功能需求

非功能需求是表示软件可信性的一种质量需求,是可信软件需求管理中至关重要的部分,是项目成败的关键。在许多软件系统,尤其是大型软件系统中,非功能需求甚至是强制的需

求,如电信系统中的性能和可靠性要求,电子商务中的安全性要求等。

一个软件系统所刻画的信息分为两类,即功能需求 (functional requirements, FRs) 和非功能需求 (non-functional requirements, NFRs)。前者描述了系统该做什么,是完成系统所指定的功能,是系统构建的主体;后者主要描述了对功能需求的约束,是一种需要分解精化的全局性软件行为的约束,是决定系统好坏的一个重要因素。

2.1 非功能需求的特征

相比于功能需求,非功能需求本身具有一些与功能性需求不同的特征,主要有以下几点:

a) 主观性。由于不具有明确的判定标准,使非功能需求具有一定的主观性,通常比较抽象。也就是说对非功能需求的判定往往根据人的主观感受来决定,不同的人对非功能需求有着不同的观点和看法。

b) 全局性。非功能需求是对系统特征的约束,通常具有全局意义,对体系结构的设计起着重要的指导作用。

c) 相关性。一个系统通常要考虑多个非功能需求,它们之间通常存在彼此的相互作用,满足一条非功能需求可能会有助或损害其他非功能需求的实现。

d) 相对性。首先,非功能需求是相对的,非功能需求的解释和重要级别很大程度上依赖于所考虑的特定系统;其次,非功能需求的实现是相对的,它的满足具有一种概念的程度。

2.2 非功能需求的定义

a) 非功能需求不是指系统的某个具体的、特定的功能行为^[5]。

b) 非功能需求不是决定系统要做什么,而是系统将怎样做^[6]。

c) 非功能需求是指软件的诸如产品的易用程度如何、执行速度如何等软件质量属性^[7]。

d) 非功能需求就是从各个角度所考虑的可能的解决方案起约束和限制作用的要求^[8]。

e) 软件的功能需求定义了一个软件期望做什么,而非功能需求则指定了关于软件如何运行和功能如何展示的全局限制^[9]。

综合上述定义,结合可信软件的特定领域的特点,本文作如下定义。

定义2 非功能需求 (non-functional requirements, NFRs) 描述的是系统开发过程中的各种软件质量属性和设计约束^[10]。

3 软件可信性经典模型的对比分析

由于软件可信性的度量以建立的模型为基础,所以,软件可信性度量模型建立过程的合理与否就成为了影响度量过程以及度量结果是否准确的关键。而目前对于软件可信性的研究,又多是在软件质量度量上进行的后续研究和改进,因此,有必要对国内外已有软件质量模型进行一个回顾和总结。

本文重点介绍已有模型中应用较广、影响较大的模型^[11-15]。其中,主要介绍七个层次型模型和两个非层次(关系型)模型,并系统地讨论它们的优点和缺点(表2)。在上述九个模型中,前五个模型分别与国内外的软件质量标准有关,是较为经典的模型;后四个则是随着新出现的技术和软件形态

而出现的比较新的质量模型。并且进一步统计了五个常用的经典模型中出现的相关属性的频次(表3)。

表2 国内外经典模型对比

模型	结构	层级数量	关系	缺点	优点
国外					
McCall	层次结构	2	多对多	成分的重叠	具有明确的度量标准
Boehm	层次结构	3	多对多	缺少相应的度量标准	考虑了硬件方面的因素
FURPS	层次结构	2	一对多	没有考虑可移植性	相对完整和全面
Dromey	层次结构	2	一对多	不全面、没有考虑可信属性	适用于不同的软件系统
ISO 9126	层次结构	3	一对多	过于宽泛、针对性不强	明确的度量标准,考虑内外部属性
Star	非层次结构	—	多对多	缺少相应的度量标准	考虑了不同角度的质量属性
BBN	非层次结构	—	多对多	缺少相应的度量标准	对质量属性予以赋权
国内					
SSC	层次结构	3	一对多	不全面、没有考虑可信属性	首次提出模型分级的概念
TRUSTIE	层次结构	3	一对多	具体度量方法不够明确	具有明确的度量标准

表3 经典模型中的相关属性

属性	模型				
	McCall	Boehm	FURPS	Dromey	ISO 9126
正确性	*				
可靠性*	*	*	*	*	*
效率*	*	*		*	*
可用性	*		*		
完整性	*				
可移植性*	*	*			*
可复用性*	*	*		*	
互连性	*				
互维护性*	*	*		*	*
可测试性	*	*			
灵活性	*	*			
功能性			*		
性能			*		
支持度			*		
易用性		*			*
功能性				*	*

通过对上述模型的归纳、对比和分析发现,存在如下两个问题:

a)它们大多是以ISO/IEC 9126模型为基础,虽然ISO模型具有完整的层次结构、具体的评价标准、综合的表达式和条件、简单和准确的定义以及各层次间一对多的关系等多种优点,但它所包括的也仅仅是通用的质量属性,并未考虑到诸如可追踪性、可访问性、可扩展性等可信属性。再加之ISO模型推出时,对软件可信性的研究尚未开始,所以,上述的模型均不能完全地适用于软件可信性度量模型的定制。

b)它们的基本原理都是一致的,即将软件质量的概念按照从上到下、从粗到细、从抽象到具体分解细化为若干层次。但这样的模型建立过程显然缺乏合理性,对于属性的选取缺乏完整性和一致性,并且属性的选取粒度过粗无法抽象。上述模型都没有将软件非功能属性作为整体进行系统的研究。

综上所述,以往的软件质量模型要么是没有考虑软件可信

性中所要求的非功能需求,要么就是在模型建立过程中对需求的分解、分类不够细致,对所有的属性都一概而论。

本文建立了一个基于关键非功能需求的软件可信性度量模型来解决上述问题。首先考虑软件可信性中起决定作用的关键非功能需求、起补充作用的非关键非功能需求以及无关的非功能需求;其次,为了更好地提取上述三类非功能需求,将它们细化为A、B、C、D四类,分别对应不同的上级非功能需求,给予四类细化后的非功能需求不同的权重,最后得到一个带有权重的非功能需求分解图(图2和3)。

4 基于关键非功能需求的非功能需求分解

4.1 关键非功能需求

虽然有许多产品特性可以称为非功能需求,但是许多系统中需要认真考虑的仅是其中的一部分,不同类别的软件、不同的应用领域,用户所关注的非功能需求不同。例如嵌入式的实时软件侧重性能和可靠性,框架应用软件看重有效性和可维护性,桌面系统强调互操作性和可用性;再比如可靠性对于航天系统是至关重要的,可维护性对于商业系统则是关键问题,完整性对于银行或者金融系统显得尤为重要,机密性对于政府或军事系统是不可忽视的非功能需求,等等。开发者必须知道哪些非功能对软件系统是关键非功能需求,对于软件系统的成功至关重要,这样他们才可能选择适当的方法来达到特定的质量目标^[16]。

定义3 关键非功能需求(critical non-functional requirements, cNFRs)是指一个软件是可信的所必需具备的非功能需求。

定义4 非关键非功能需求(non-critical non-functional requirements, ncNFRs)是指那些除了必须具备的关键非功能需求之外可能还需要具备的其他非功能需求。

进而,将非功能需求细分为A、B、C、D四类,以更方便地提取可信性需求,并建立各类可信性需求关联度以便更好地对软件可信性进行度量。

A类非功能需求(NFR_A)是基本非功能需求,反映各类软件共同的基本非功能要求的需求,是必须选择的非功能需求。无特殊情况时,一般的软件都应选取。将其关联度R定义为1.0。

B类非功能需求(NFR_B)是可选非功能需求,反映各类不同软件的可选非功能要求的需求,它的选取是与某类(些)特定软件的非功能要求相关。与A类的关联度为 $0.5 < R_B \leq 0.75$ 。

C类非功能需求(NFR_C)是非常规可选非功能需求,反映不常使用的非功能需求,它的选取可以是评价委托单位特别要求进行评价的非功能需求,或者是从不同用户角度出发的非功能需求。与A类的关联度为 $0.25 < R_C \leq 0.5$ 。

D类非功能需求(NFR_D)是无关非功能需求,反映与所评价的软件无关的非功能需求,一般不予考虑。与A类的关联度为 $0 < R_D \leq 0.25$ 。

A类非功能需求应尽可能地选用;根据软件的特点选取B类非功能需求;根据特定的要求和评价需求选取C类非功能需求;对于D类非功能需求视情况可以予以适度考虑。

4.2 非功能需求分解

软件可信性是软件非功能需求的一个关键子属性,是可信

软件必备的属性,其他诸如技术属性、经济社会因素可以在具体系统中具体考虑(图2)。

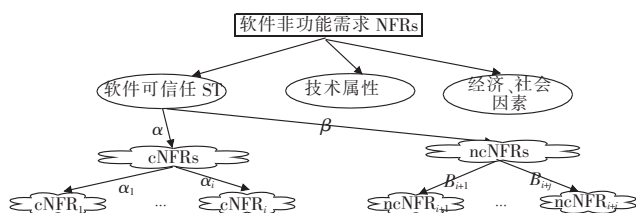


图2 非功能需求分解图1

本文中只考虑基于软件可信性的评估问题。而软件可信性又可以进一步细分为关键非功能需求和非关键非功能需求,其中,将A类和B类非功能需求划分为关键非功能需求,将C类非功能需求划分为非关键非功能需求(图3)。

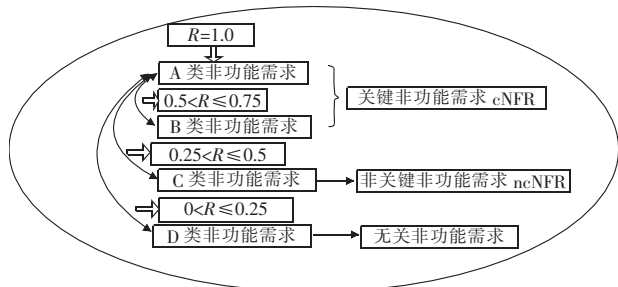


图3 非功能需求分解图2

假设软件可信性(ST)精化为 $i+j$ 项非功能需求(NFRs),分别为 $NFR_1, \dots, NFR_i, NFR_{i+1}, \dots, NFR_{i+j}$,其中前 i 项为cNFRs,后 j 项为ncNFRs(图2)。假设所有cNFRs组成的集合在所有NFRs中所占的权重为 α ,所有ncNFRs组成的集合在所有NFRs中所占的权重为 β ,满足 $0 \leq \beta < \alpha \leq 1, \alpha + \beta = 1$ 且 $\alpha > 0.5 > \beta$;假设cNFR $_m(m=1, \dots, i)$ 在整个cNFRs中所占的权重依次为 $\alpha_1, \dots, \alpha_i$,且满足 $0 \leq \alpha_1, \dots, \alpha_i \leq 1, \sum_{m=1}^i \alpha_m = 1$;ncNFR $_n(n=i+1, \dots, i+j)$ 在整个ncNFRs中所占的权重依次为 $\beta_{i+1}, \dots, \beta_{i+j}$,且满足 $0 \leq \beta_{i+1}, \dots, \beta_{i+j} \leq 1, \sum_{n=i+1}^{i+j} \beta_n = 1$ (表4)。

表4 非功能需求权重表

软件可信性 ST						
cNFRs			ncNFRs			
	cNFR ₁	...	cNFR _i	ncNFR _{i+1}	...	ncNFR _{i+j}
权重	α_1	...	α_i	β_{i+1}	...	β_{i+j}
	$\sum_{m=1}^i \alpha_m = 1$			$\sum_{n=i+1}^{i+j} \beta_n = 1$		
	$\alpha + \beta = 1$					

5 基于关键非功能需求的软件可信性度量模型

对于关键非功能需求,只有在它们必须全部具备的条件下才能谈软件可信的程度,而这其中只要有一个性质没有满足,那么该软件就为不可信。对于一个给定的软件,只要求用户设定一个阈值 y_0 ,如果任何用来刻画软件可信性的非功能需求的属性值低于该阈值,就认为软件不可信;只有在所有非功能需求的属性值均大于由用户设定的阈值 y_0 的情况下,才可以考虑软件可信性的评估问题。

在所有非功能需求均满足用户初始阈值 y_0 的情况下,对于具体软件的具体领域、具体利益相关者,综合考虑软件可信性的实际状况,寻找出软件所关注的关键非功能需求,采用主

客观赋权法对各个关键非功能需求进行权重分析及修正。再依据完整性、简明性、主要性、层次性、可比性等原则,在概念清楚、数据易得、各自独立的基础上对选取的各个关键非功能需求作进一步的分解精化,找出所对应需求项的最基本组成单元,构建元需求集;结合元需求的属性值,对软件可信性的评估建立综合的评价方法,为软件可信性的分析及管理提供理论及模型支持,建立可信软件度量的技术标准或管理标准方案。基于关键非功能需求的软件可信性度量模型如图4所示。

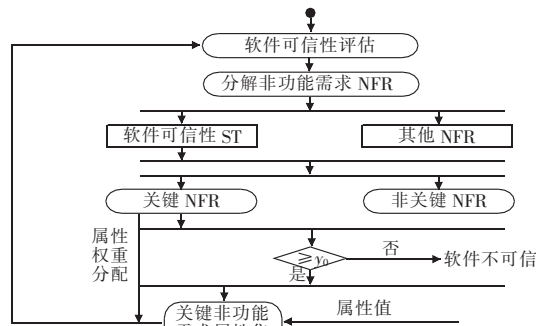


图4 基于关键非功能需求的软件可信性度量模型

软件可信性可以看做是软件系统特性的综合度量和评价的过程,但绝不仅仅是简单的线性求和。在实际应用环境中,在评估软件可信性时,由于若干非功能需求并不是一概而论,各个非功能需求的重要程度取决于系统的实际需求、开发成本和外在环境等诸多因素,并不能用一个单一的标准来衡量,而是需要一个具有多方面指标的测度。因此,软件可信性是一个多目标的决策问题,可以看做是在各个非功能需求度量基础上的函数:

$$P_{ST} = f(P_1, P_2, \dots, P_n, \omega_1, \omega_2, \dots, \omega_n) \quad (1)$$

其中: P_{ST} 表示一个软件系统的可信性; $P_1, P_2, \dots, P_n$ 表示软件非功能需求的度量值; $\omega_1, \omega_2, \dots, \omega_n$ 表示各个非功能需求的权重分配结果。这样,系统在不同的设计和配置下,可以对它的非功能需求作出定量的比较和分析,对各需求的强调程度与侧重点的不同直接影响到系统的可信性。

在本文中,由于对非功能需求的分类,可以将式(1)作进一步改进,加入A、B、C、D四类非功能需求的关联度以及关键非功能需求和非关键非功能需求的相对权重,得到式(2)。

$$P_{ST} = P_{cNFRs} + P_{ncNFRs} = \sum f(P_i, R_i, w_i) \quad i = A, B, C, D \quad (2)$$

其中: P_{ST} 表示一个软件系统的可信性; P_i 表示各类非功能需求的度量值; R_i 表示各类非功能需求的关联度; w_i 表示各类非功能需求的权重分配结果。

6 实例分析

本文选取了Web软件作为本文的实例分析,是因为Web软件是在互联网技术快速发展背景下,软件形态向网络化方向转变趋势下诞生的新型软件系统,Web软件正以其开放性等特点成为互联网资源、技术与方法聚合的完美结合与体现的优秀典范。但同时由于其运行环境的不确定性与复杂性,Web软件可信性也是Web软件产品成功的关键因素,所以,对Web软件可信性进行度量是亟待解决的问题。其中,“在线旅行”网站伴随着互联网的不断普及、支付方式的日益完善以及在线企业对服务、产品和用户体验的不断提升,也成为用户参与体

验度较高的 Web 软件的优秀代表之一,所以,对“在线旅行”软件进行的可信评估有其实用意义。

依照国际上关于软件可信性的经典模型(表3)中分别提及的软件非功能需求,将出现频次最多(frequency $i \geq 3$)的非功能需求——可靠性、效率、可移植性、可复用性、可维护性作为 A 类非功能需求($R_A = 1.0$)。具体的非功能需求释义可分别查阅相应参考文献[11~15]。因文章篇幅有限,这里将不作赘述。

根据 Web 软件的特点,Behkamal 等人提出的 Web 软件属性图中包括安全性、可扩展性、效率、可访问性等非功能需求;Nabil 提出的 Web 软件概念模型中主要包括可追踪性、可定制性、易恢复性等非功能需求;罗新星等人^[4]考虑 Web 软件外部使用满意度后提出基于 ISO 的 Web 软件非功能需求模型,主要包括安全性、可追踪性、可访问性、可定制性、可导航性等非功能需求。据此,选取 B 类非功能需求——安全性、可追踪性、可访问性以及可扩展性,取其关联度为 $R_B = 0.75$ 。

从“在线旅行”网站特别要求的非功能需求出发,选取可导航性及易恢复性为 C 类非功能需求,取其关联度为 $R_C = 0.5$ 。

此处,因为 D 类非功能需求的关联度 R 很小,故可以忽略不计。

在确定了 Web 可信性中的关键非功能需求后,需要对各个关键非功能需求给予初始阈值,这里笔者认为,已经提取的关键非功能需求均符合 Web 软件的特性。权重和非功能需求的属性值获取不是本文的研究重点,但为了更具体地演示软件可信性度量模型的全过程,本文采取等分的方法。设 A 类可信性需求权重为 0.33, P (可靠性,效率,可移植性,可复用性,可维护性) = (0.2, 0.2, 0.2, 0.2, 0.2), $R_A = 1.0$; B 类可信性属性需求权重为 0.33, P (安全性,可追踪性,可访问性,可扩展性) = (0.25, 0.25, 0.25, 0.25), $R_B = 0.75$; C 类可信性需求权重为 0.33, P (可导航性,易恢复性) = (0.5, 0.5), $R_C = 0.75$ 。所以“在线旅行”网站的可信性为 $0.33 \times 1 \times 0.2 \times 5 + 0.33 \times 0.75 \times 0.25 \times 4 + 0.33 \times 0.5 \times 2 = 0.7425$ 。“在线旅行”Web 软件的可信性度量模型如图 5 所示。

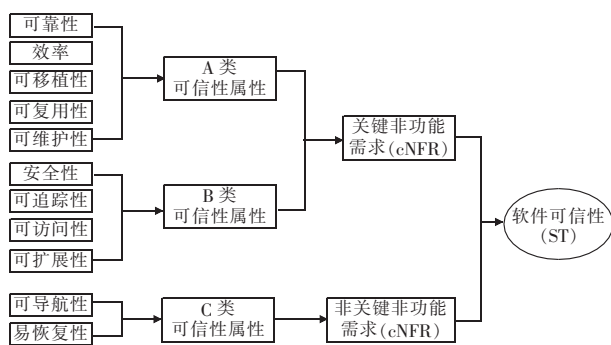


图5 “在线旅行”Web软件的可信性度量模型

7 结束语

软件的非功能需求对于保证一个软件系统的质量具有重要意义,软件可信性又是软件非功能需求中重要的组成部分,而非功能需求的特性导致其不易被捕捉和分析,从而造成研究上的困难。因此,如何在非功能需求的基础上对软件的可信性进行评价将是研究的重要课题之一。

基于这样一个事实,本文给出了一个基于关键非功能需求的软件可信性度量的模型,并详细分析了非功能需求的分解、分类、软件可信性中的关键非功能需求的获取、软件可信性度量的流程。此外,还对软件可信性中分解出的关键非功能需求和非关键非功能需求的权重予以解释。本文提出的模型是一个通用的度量软件可信性的模型,适用于不同形态的软件,支持软件可信性、可信证据、评估指标的定制,对于软件可信性的度量具有一定的指导意义。

本文的工作虽然提出了一个度量软件可信性的模型,但是此模型还需要进一步地定性定量的系统分析。在下一步工作中,笔者将研究需求属性值的获取及分析、非功能需求的权重及主客观赋权,并试图应用到软件开发的早期过程中,使得后期软件产品符合人们预期的可信性,在软件开发的整个生命周期中从总体上对软件可信性进行研究;此外,各个非功能需求之间的权衡模型也将是笔者未来的工作之一。

参考文献:

- [1] 刘克,单志广,王戟,等.可信软件基础研究重大研究计划综述[J].中国科学基金,2005,2(3):145-151.
- [2] 罗新星,王卫芳.基于本体的元需求模型及元需求研究[J].计算机应用研究,2012,29(4):1391-1394.
- [3] 刘旭东,郎波,谢冰,等.软件可信分级规范[R].北京:可信的国家软件资源共享与协同生产环境课题组,2009.
- [4] 罗新星,朱名勋.可信软件中非功能需求 FO-QSIG 冲突权衡模型[J].系统工程,2010(2):101-105.
- [5] BROWN J R, KASPAR H, LIPOW M, et al. Characteristics of software quality[M]. Amsterdam: North-Holland Publishing Company, 1978.
- [6] EVANS M W, MARCINIAK J J. Software quality assurance and management[M]. New York: Wiley, 1987.
- [7] WIEGERS K E. 软件需求[M]. 陆丽娜,王忠民,王志敏,译.北京:机械工业出版社,2000.
- [8] CHEVLIN D H. Software requirements: definition and specification[J]. Biomedical Instrumentation Technology, 1996, 30(2): 150-152.
- [9] PAECH B, KERKOW D. Non-functional requirements engineering-quality is essential[C]//Proc of the 10th International Workshop on Requirements Engineering Foundation for Software Quality. 2004.
- [10] ZHU Ming-xun, LUO Xin-xing, CHEN Xiao-hong, et al. A non-functional requirements tradeoff model in trustworthy software[J]. Information Sciences, 2012, 191(5): 61-75.
- [11] FITZPATRICK R. Software quality: definitions and strategic issues[R]. [S. l.]: Dublin Institute of Technology, 1996.
- [12] BOEHM B W, BROWN J R, LIPOW M. Quantitative evaluation of software quality[C]//Proc of the 2nd International Conference on Software Engineering. [S. l.]: IEEE Computer Society, 1976: 592-605.
- [13] KAN S H. 软件质量工程——度量与模型[M]. 王振宇,陈利,余扬,译.北京:机械工业出版社,2003.
- [14] ABRAN A, KHELIFI A, SURYN W, et al. Usability meanings and interpretations in ISO standards[J]. Software Quality Journal, 2003, 11(4): 325-338.
- [15] DROMY R G. A model for software product quality[J]. IEEE Transactions on Software Engineering, 1995, 21(2): 146-162.
- [16] 陶红伟.基于属性的软件可信性度量模型研究[D].上海:华东师范大学,2011.