

基于质量功能展开的可信软件需求映射方法

熊 伟, 王晓暉

(浙江大学 管理学院, 浙江 杭州 310058)

摘 要: 为了保证和提高软件的可信性, 针对当前可信软件需求管理在将可信性需求向软件设计过程映射方面存在的不足, 提出基于质量功能展开(QFD)方法的可信软件需求映射方法. 该方法在用数量化理论 3 类(QM3)方法定量分析可信软件需求的基础上, 借助 QFD 方法中的质量屋(HOQ)矩阵, 将可信性需求映射到软件设计开发的整个过程, 并在过程中追踪和控制这些需求, 以使其在软件开发过程中得到正确而一致的实现, 从而系统化地保证软件的可信性. 用某金融软件系统的开发实例进行验证, 说明所提方法能有效地管理和控制可信性需求, 对提高软件系统的可信性和保证软件质量具有重要意义.

关键词: 可信软件; 需求管理; 质量功能展开(QFD); 质量屋(HOQ); 数量化理论 3 类(QM3)

中图分类号: TP 311

文献标志码: A

文章编号: 1008-973X(2010)05-0881-06

Method for mapping software dependability requirements based on quality function deployment

XIONG Wei, WANG Xiao-tun

(School of Management, Zhejiang University, Hangzhou 310058, China)

Abstract: Aiming at the dependability requirements to architecture mapping in software engineering, a requirements mapping method based on quality function deployment (QFD) was proposed to assure and improve the software dependability. The dependability requirements were deployed into the entire development process by combining with the quantitative analysis of dependability requirements using quantification method of type 3 (QM3) and the house of quality (HOQ). And the requirements can be correctly and consistently fulfilled in the software development process by tracing and controlling the requirements using HOQ. A case study of a financial software development was used to illustrate the method. Results show that the method can effectively enhance the dependability of software system and assure the software quality.

Key words: dependable software; requirements management; quality function deployment (QFD); house of quality (HOQ); quantification method of type 3 (QM3)

随着社会的进步和计算机技术的高速发展, 软件已渗透到人类生活和生产的方方面面, 人们越来越依赖软件. 然而, 软件的生产现状和质量一直不能令人满意^[1], 尤其是应用于航空航天、核电及国防等领域的安全关键软件系统, 其失效常常会对人类和环境造成严重的乃至灾难性的后果. 1996 年, 欧洲

航天局首次发射阿丽亚娜 5 号火箭失败, 导致经济损失 25 亿美元, 其直接原因就是火箭控制系统的软件故障. 在我国, 2006 年中航信离港系统发生的软件系统故障导致了近百个机场登机系统瘫痪. 这类由于软件缺陷而导致的重大灾难和事故已不胜枚举, 而造成的影响也越来越大. 据美国国家标准技术

研究院估算,美国每年因软件失效所造成的年度经济损失近 600 亿美元,约占其 GDP 的 0.6%^[2]。鉴于此,人们对软件的可信性提出了更高的要求,软件的可信性已成为软件工程的重要研究方向和国际上致力解决的重大问题^[2-4]。

为了保证和提高软件可信性,如何进行可信性需求建模向设计过程的映射,将可信性需求映射到软件设计开发的整个过程,并在过程中追踪和控制这些需求,以使其在软件开发过程中得到正确而一致的实现,是亟待解决的关键问题。此外,软件需求工程的发展,使人们认识到只有最终用户直接参与并发挥主导作用,才能真正解决问题空间与求解空间的一致性,消除计算机领域和应用领域之间的鸿沟^[5]。作为用户导向的产品设计方法,质量功能展开(quality function deployment, QFD)^[6]是连接用户与技术人员的有效方法。QFD 的特点是以顾客需求为依据,通过质量屋(house of quality, HOQ)将顾客需求转化为产品开发设计过程的一系列技术要求,在开发初期就对产品的质量和适用性实施了全方位的保证。QFD 在商业软件开发中已有不少应用案例^[6-7],它被认为是日本式质量管理的特点,已在美国、日本等许多国家得到广泛应用。

针对当前研究在可信性需求映射机制方面存在的不足,本文借助 QFD 理论中的 HOQ 矩阵,结合数量化理论 3 类(quantification method of type 3, QM3)的定量解析功能,提出一种基于 QFD 的可信软件需求映射方法,以在软件设计过程中跟踪和控制可信软件需求,进而提高软件的可信性。

1 可信软件及其需求管理现状

1.1 软件可信性

可信性是在正确性、可靠性、安全性、完整性、可用性等众多概念的基础上发展起来的一个新的综合性概念^[3]。对于可信性的定义,国内外学者至今尚未形成共识。例如, Littlewood 等^[3]认为可信性应包含可靠性、可靠安全性、保密安全性和可用性。Avizienis 等^[8-9]将可信性定义为可用性、可靠性、安全性、机密性、完整性和可维护性的综合。此外,美国可信计算组织(TCG)及国际信息处理协会(IFIP)等关于可信性的定义均有不同的表述^[10-11]。陈火旺等^[1, 12]给出了各自对于可信性的定义。不同学者虽常采用不同的定义和分类,但对可信软件的基本特征却有共识,即软件系统能够持续地提供可信赖的服务,其服务的整个过程和结果与人的预期相符,在

受到干扰时仍能提供连续的服务,以避免外部干扰给用户带来的损失。

对于“可信软件需求”,暂无权威定义。Donzelli 等^[13]指出可信需求是因环境而变的,其具体含义在很大程度上依赖于利益相关者所处的环境。环境不同,利益相关者所关注的特征也不同。Donzelli 等^[13]以“可靠性、可用性、安全性、机密性和生存性”作为可信需求的主要特征,并以此为基础提出了可信性的统一模型(unified model of dependability, UMD),以帮助不同人员获取、交流,并理解可信需求。笔者认为“可信软件需求”是一般的软件需求中体现“可靠性、可用性、安全性和生存性”等可信性质的软件需求。可信软件需求包含于一般软件需求,前者是后者的子集。本文将“可靠性、可用性、安全性、保密性及容错性”这 5 个属性作为关注焦点,研究基于 QFD 的可信软件需求映射方法。值得注意的是,本文的研究结果并不依赖于可信性属性的选择,对于“生存性、实时性和完整性”等同样适用。

1.2 可信软件需求管理现状

高可信软件系统研发的关键之一是面向可信性的需求分析,需求分析作为软件生命周期的首个阶段,是后续所有软件设计活动的基础,贯穿于整个软件生命周期。需求工程已成为软件领域的研究热点之一^[14]。然而,现有的需求工程方法偏重于需求分析和建模的研究,较少论及如何将需求映射到软件开发过程,并保证其在开发过程中得到正确而一致的实现。至今需求工程仍是软件工程中最复杂、困难的过程之一。

可信软件的需求工程更是当前面临的新问题。可信性需求对于软件系统是非常重要的软件规约要素,当前对计算机系统的可信性需求已从单属性需求扩展为多属性综合需求。另一方面,需求映射(requirements mapping)是确保用户需求在软件开发中得到考虑的最有效方法^[15-16]。Xu 等^[17]认为应该将复杂的用户要求映射到体系架构的设计中去,需求映射是解决软件可信性问题的方法。

从当前的研究与实践来看,可信软件的需求管理中仍存在下列主要问题。

1) 可信性需求获取的困难性。当前的需求工程中需求抽取阶段虽有问卷法、面谈法、情景实例法等基于目标的方法,以及场景分析法、卡片分类法等基于模型的知识工程方法^[5, 14],但它们仍不能对领域本体进行精确描述,也不能充分分析和定义一致的可信性需求。将可信性需求分解,并转化为对软件行为的约束是定义与实现可信性需求的关键。

2) 可信性需求分析的不完整性. 需求分析是指从用户的原始描述中抽取对预开发软件的功能和可信性等方面的要求, 并加以分析的过程. 通常人们对事物的认识很难一次到位, 用户对软件的需求也是一个不断进化的过程, 这导致一开始需求很难被充分挖掘. 需求的不完整性是软件开发失败的主要原因之一.

3) 可信性需求建模的复杂性. 软件需求中常混杂着各种各样的需求, 且各个需求项目的概念范畴也不相同. 此外, 非功能性需求与功能性需求之间错综复杂的联系, 以及当前对非功能需求分析建模技术的缺乏等都大大增加了可信性需求建模的复杂性.

1.3 问题解决思路

1) 可信性需求分析过程可视化. 将包括可信需求抽取活动在内的需求分析过程可视化, 并建立相应的可视化模型. 需求抽取及分析过程偏向创造性思维活动, 其过程难以让第三者理解, 因而可视化的需求分析过程有助于用户与设计者之间的沟通和协调. 为此, 把 QFD 技术应用到可信软件需求分析中, 借以改善需求获取的困难性. QFD 方法通过系统地展开用户需求与技术特性之间的关系来保证产品设计质量. 利用 HOQ 矩阵可使需求映射的全过程可视化, 便于各部门的人员交流和理解^[18].

2) 可信性需求分析的充分化. 如何在调查的初期阶段提高可信性需求分析的充分性和完整性, 并有效地管理需求的进化是解决问题的关键. 软件工程中的迭代和演化模型提供了可行的解决思路, 但现有方法的迭代和演化基准并不十分明确. QM3 与其他 3 种方法(数量化理论 1 类、2 类、4 类)都是数据分析类的统计方法^[19]. Xiong 等^[18]提出的在需求分析过程中反复应用 QM3 的方法也是一种迭代和演化的过程, 它能客观地使需求项目层次化, 同时也使需求得以充分化和完整化, 更重要的是它给出了迭代和演化的定量依据.

3) 可信性需求的结构化. 结构化过程是减少复杂性的过程, 因而将需求结构化可降低需求建模的复杂性. 但现有的方法, 例如, 结构化分析和原型化方法, 由于它们在结构化过程中的基准不十分明确, 导出的结果容易依赖于开发者的经验; 而基于 QM3 的方法正是实现结构化的定量方法, 其结果更客观.

2 可信软件需求映射方法

首先应用 QFD 技术获取用户对可信软件系统的原始需求, 并用亲和图法^[19]整理, 按 QFD 中质量需求的表述规范定义可信需求形式规约; 其次用

HOQ 进行可信需求分析过程的可视化, 并构造可视化模型; 然后用 QM3 进行定量解析. 运用 HOQ 把可信性需求 $R_{Di}(i=1, \dots, m)$ 有效地转换成技术特性 $T_j(j=1, \dots, n)$, 并通过一系列 HOQ 逐层顺次展开, 参考可信需求重要度优先级排序, 将可信性需求逐步映射至软件设计开发的全过程、以及子系统和功能模块, 建立起一系列流程和控制目标, 通过实现和协调这些目标, 最大限度地实现可信性需求.

可信软件需求映射的流程如图 1 所示. 其具体步骤如下.

1) 用 QFD 技术获取客户原始需求. 通过市场调查和深度访谈, 以及查阅投诉报告等历史数据来获取用户对可信软件的原始需求数据. 同时, 召集部分用户集会, 通过 QFD 理论的自由发想法^[19], 并以各人填写卡片的形式, 提出他们对目标软件的需求.

2) 原始信息的变换与整理. 逐一探讨原始数据及卡片, 并转换成只含有一个具体含义的语言信息. 用 QFD 理论的亲和图法整理语言信息并分类, 对已经整理好的一般需求(包含功能需求和非功能需求), 再参照可靠性、可用性、安全性、保密性和容错性等可信性特征, 探讨 $R_{Di}(i=1, \dots, m)$. 然后逐个抽出次级可信需求, 构造可信软件需求展开表. 同时邀请用户评价各项需求的重要度 $w_i(i=1, \dots, m)$.

3) 数据项目的抽取和整理. 探讨某一可信需求需要怎样的输入和输出数据, 逐一抽取数据项目, 构造软件数据展开表.

4) 可信软件需求-数据 HOQ 的构造. 整理可信

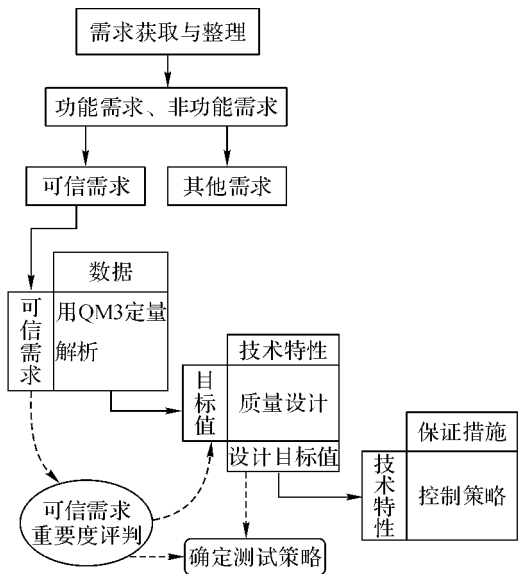


图 1 基于 QFD 的可信软件需求映射方法

Fig. 1 Method for mapping software dependability requirements based on QFD

软件需求项目及其对应的数据项目,把可信软件需求和数据分别作为 HOQ 的输入和输出变量,探讨两者的相关关系,构造出可信软件需求-数据 HOQ.

5) 运用 QM3 的定量结构化分析. 运用 QM3 理论,使需求与数据关系聚类,形成结构化的 HOQ 矩阵. 在分解过程中,通过调整子系统的水平(层次)可以逐渐得到完全结构化的需求数据关联表. 这些操作过程本身就是软件结构分析过程,其结果客观地导出了不依存于人的经验的软件层次结构. 通过定量分析使软件需求项目层次化.

6) 可信软件需求-技术特性 HOQ 的构造. 考察实现可信软件需求所必需的技术特性 $T_j (j=1, \dots, n)$, 建立质量设计矩阵. 通过可信软件规划 HOQ 将可信软件需求 R_{Di} 转化成对应的技术特性 T_j , 同时 R_{Di} 的重要度映射给了 T_j , 而 T_j 的重要度为

$$w(T_j) = \sum_{i=1}^m w_i R_{Di}. \quad (1)$$

式中: $i=1, \dots, m, j=1, \dots, n, w_i$ 为用户需求 R_{Di} 的重要度, R_{ij} 为用户需求-技术特性相关系数. 求出 T_j 的重要度后,结合技术可行性和成本因素,则可确定最终的可信软件 T_j 目标值.

7) 建立测试和控制策略矩阵. 将上述 HOQ 的输出作为“技术特性-保证措施”HOQ 的输入,探讨 T_j 与保证措施 $A_{Mk} (k=1, \dots, s)$ 之间的相关关系 R_{jk} , 则可将 T_j 的重要度以及目标值映射给 A_{Mk} , 并根据式(2)确定 A_{Mk} 的重要度:

$$w(A_{Mk}) = \sum_{j=1}^n w(T_j) R_{jk}. \quad (2)$$

式中: $j=1, \dots, n, k=1, \dots, s$. 根据 $w(A_{Mk})$, 结合技术和成本等因素,则可确定需要重点保证的措施. 在此基础上可建立相应的控制策略矩阵,确立控制人员、控制目标等.

8) 如此继续下去,将可信软件需求逐步映射到软件设计开发的全部过程及软件整个生命周期的各个环节,建立起一系列流程和控制目标,通过实现和协调这些目标,最大限度地满足用户的可信性需求.

3 实例研究

H 公司是一家专门从事软件开发的高新技术企业. 本文所提的可信软件需求映射方法已成功地应用于 H 公司某金融软件系统的开发. 用户对金融软件系统的可信性有较高要求,能很好地检验本文所提方法在实际的可信软件开发中的有效性. 由于该金融软件系统开发项目规模较大,涉及的用户需

求和技术特性众多. 为突出重点,本文选取“PO 状态检索”模块来说明所提方法的应用.

1) 用 QFD 技术获取客户原始信息. 首先,经过深入咨询和调查后,获取客户对软件的原始需求,得到“能够检索 HF5 系统中的 PO 信息”、“画面能够显示出全部项目”等 45 项原始需求.

2) 原始信息的变换与整理. 对这些原始需求,运用亲和图法进行整理. 将顾客需求分类整理成“功能需求和非功能需求”. 继而对照可信性属性,从中细分出可信性需求. 对于本案例,可信需求主要包括“保密性和容错性”. 具体而言,二次水平需求“用户登录”所包含的三次水平需求“保证密码安全性”、“设置用户权限”和“执行身份验证”是典型的保密性需求,而二次水平需求“防错性”所涵盖的三次水平需求“错误检查”和“有错误提示对话框”则是容错性需求,继而邀请用户评价各项可信需求的重要度.

3) 数据项目的抽取和整理. 根据整理过的需求,逐一抽出相应的输入、输出数据. 例如,从保密性需求“保证密码安全性”可以抽出“密码库”和“用户身份”等数据项目.

4) 可信软件需求-数据 HOQ 的构造. 通过探讨数据与软件需求项目的相关关系,构建软件系统的需求-数据 HOQ.

5) 运用 QM3 的定量结构化分析. 对上述 HOQ 运用 QM3 理论进行解析,参照求得特征根和特征向量(限于篇幅,此处不再列出),将软件需求定量地分解为“检索”、“用户登录”和“防错性”等几部分,如图 2 的 3 块灰色底纹区域所示. 对“PO 状态检索”模块,反复应用 QM3,并利用需求项目间的数量关系(特征向量),定量地逐步导出“PO 状态检索”模块的需求层次结构,如图 2 左半部分所示.

6) 可信软件需求-技术特性 HOQ 的构造. 探讨 R_{Di} 与 T_j 之间的相关关系,建构用户需求-技术特性 HOQ,如图 3 所示. 可知,保密性需求“执行身份验证”重要度为 8.5,在所需求中是最重要的. 在此 HOQ 中,通过两者之间关系的强弱, R_{Di} 的重要度就可以转化成 T_j 的重要度. T_j 目标值的确定也参照了 T_j 的重要度,而这种重要度的转换和传递,可以一直延续到编码或其他开发阶段,有助于把握软件设计的重点与核心.

据式(1)可求得 T_j 的重要度,在图 3 中 T_j “错误检测”的重要度最高. 在编码和过程控制中需要特别重视这一技术特性,因为它直接影响到所开发软件系统的可信性和软件质量. 其他如“登陆按钮属性、密码管理和提示信息”等,由于其重要度较高,在

用户需求			PO信息											
			一次	检索					用户登录					防错性
			二次											
			三次	发票信息	PO 明细表	PO 信息一览表	与 HFS 系统接口	检索类型	密码库	用户身份	权限类型	错误类型	交互方式	
P O 状态检索	检索功能	检索发票		◎				△						
		检索PO明细			◎			△						
		检索PO一览				◎		△						
		检索HFS系统中的PO信息					◎	△						
		设定检索PO关键词		△	△	△	△	◎						
	用户登录	保证密码安全性								◎				
		设置用户权限									△	◎		
		执行身份验证								△	◎	△	△	
	防错性	错误检查						○				△	◎	△
		有错误提示对话框												◎

◎-强相关；○-中相关；△-弱相关；空白-不相关

⊙-强相关；○-中相关；△-弱相关；空白-不相关

图 2 用户需求-数据质量屋(局部)

Fig.2 User needs-data HOQ (partial)

用户需求			用户登录					检索结果处理			防错性	
			密码管理	设定用户权限	登录按钮属性	报表生成器	保存按钮属性	打印设置	画面转换关系	错误检测	提示信息	用户需求重要度
PO 状态检索	用户登录	保证密码安全性	⊙		○							4.4
		设置用户权限		⊙	△							4.4
		执行身份验证	○		⊙					△	△	8.5
	检索结果处理	以指定文件格式保存发票				⊙	⊙					7.1
		能输出检索结果				△		⊙	○			4.4
		按要求显示检索结果							⊙			4.4
	防错性	执行错误检查			△					⊙		4.4
		有错误提示对话框			△					○	⊙	4.4
	技术特性重要度		30	13	47	26	21	13	22	57	26	
	技术竞争性评估	本项目	4	4	5	5	5	4	5	5	4	

⊙-强相关；○-中相关；△-弱相关；空白-不相关

图 3 用户需求-技术特性质量屋(局部)

Fig.3 User needs-technical characteristics HOQ (partial)

开发中应重点关注.

7) 建立测试和控制策略矩阵. 首先探讨 R_{jk} , 构建“技术特性-保证措施”HOQ (限于篇幅, 此处从略), 则由式(2)可确定 $w(A_{Mk})$. 对于最重要的一些管理控制措施, 应优先保证实施. 根据技术特性的不同性质以及不同的重要度, 确定不同评估标准、方法和实施人员, 制定出控制策略矩阵, 如表 1 所示. 表 1 给出了相应的评估标准、评估方法和具体的实施者等信息, 为在开发过程中保证软件的可信性和提高软件质量指明了方向.

表 1 控制策略矩阵(局部)

Tab.1 Control strategy matrix (partial)

过程特性	管理控制项目	软件质量保证措施		
		评估标准	评估方法	实施者
状态检索实现过程	版面规格	客户要求	同行评审	评审小组
	画面代码评审	每千行源代码故障数	抽样检查	PL、PG、SQA
		大于 12 个		
		大于 12 个		
	功能代码评估	每千行源代码故障数	抽样检查	PL、PG、SQA
		大于 10 个		
		大于 10 个		
	测试密度	每千行源代码查找	测试计划	项目组、测试员
		100 个故障		
		100 个故障		

总之, 本方法通过对可信软件需求的分析, 将可信软件需求通过层层映射, 建立可信需求与软件设计开发全过程中各个环节的对应关系, 实现对可信软件需求的过程管理, 从而确保可信需求的完整实现. 本方法也是实现预防式质量控制的过程. 从后续的用户反馈信息得知, 用户需求得到了很大程度的满足, 软件的质量和可信性也得到了有效保证.

4 结 语

本文提出了一种基于 QFD 的可信软件需求映射方法, 该方法以 HOQ 系列矩阵为工具, 以 QM3 的定量解析为基础, 确定可信软件需求向软件设计开发过程映射的方向, 使技术人员可以在软件开发的全过程中追踪、控制和实现这些可信性需求. 本文所提方法是面向用户满意的可信软件需求管理的重要支撑技术. 通过在某金融软件开发项目中的实际应用, 表明该方法为可信软件需求向设计开发过程的映射提供了有效的途径, 从而为在软件设计过程中对可信软件需求实施全方位保证提供了支持.

本文对可信软件需求的管理进行了初步探索, 但由于案例所限, 仅考查了保密性和容错性 2 个可信性质. 进一步的研究应该重点关注应用于航空航天、核电和国防建设等关键领域的安全软件, 充分挖掘软件的可信性需求, 研究可信性需求的冲突管理及过程控制模式问题.

参考文献(References):

- [1] 陈火旺,王戟,董威. 高可信软件工程技术[J]. 电子学报, 2003, 31(12A): 1933-1938.
CHEN Huo-wang, WANG Ji, DONG Wei. High confidence software engineering technologies [J]. **The Chinese Journal of Electronics**, 2003, 31(12A): 1933-1938.
- [2] 刘克,单志广,王戟,等. “可行软件基础研究”重大研究计划综述[J]. 中国科学基金, 2008, 22(3): 145-151.
LIU Ke, SHAN Zhi-guang, WANG Ji, et al. Overview on major research plan of trustworthy software [J]. **Bulletin of National Natural Science Foundation of China**, 2008, 22(3): 145-151.
- [3] LITTLEWOOD B, STRINGINI L. Software reliability and dependability: a roadmap[C] // **Proceedings of the Conference on the Future of Software Engineering**. Limerick: ACM, 2000: 175-188.
- [4] AVZIENIS A, LAPRIE J C, RANDELL B, et al. Basic concepts and taxonomy of dependable and secure computing [J]. **IEEE Transactions on Dependable and Secure Computing**, 2004, 1(1): 11-33.
- [5] SAIEDIAN H, DALE R. Requirements engineering: making the connection between the software developer and customer [J]. **Information and Software Technology**, 2000, 42(6): 419-428.
- [6] CHAN L K, WU M L. Quality function deployment: a literature review [J]. **European Journal of Operational Research**, 2002, 143(3): 463-497.
- [7] HAAG S, RAJA M K, SCHKADE L L. Quality function deployment usage in software development [J]. **Communications of the ACM**, 1996, 39(1): 41-49.
- [8] AVZIENIS A, LAPRIE J C, RANDELL B, et al. Basic concepts and taxonomy of dependable and secure computing [J]. **IEEE Transactions on Dependable and Secure Computing**, 2004, 1(1): 11-33.
- [9] LAPRIE J C. **Dependability: basic concepts and terminology, dependable computing and fault tolerance** [M]. New York: Springer-Verlag, 1992: 5-13.
- [10] Trusted Computing Group. TCG specification architecture overview [EB/OL]. [2008-05-27]. http://www.trustedcomputinggroup.org/groups/TCG_1_4_Architecture_Overview.pdf.
- [11] IFIP—International Federation for Information Processing WG-10.4. [EB/OL]. [2009-08-22]. <http://www.dependability.org/wg10.4/>.
- [12] 杨仕平,熊光泽,桑楠. 安全关键实时系统高可信集成技术的研究[J]. 电子学报, 2003, 31(8): 1237-1241.
YANG Shi-ping, XIONG Guang-ze, SANG Nan. Research on high dependability integration technology of safety critical real time systems [J]. **The Chinese Journal of Electronics**, 2003, 31(8): 1237-1241.
- [13] DONZELLI P, BASILI V. A practical framework for eliciting and modeling system dependability requirements: experience from the NASA high dependability computing project [J]. **Journal of Systems and Software**, 2006, 79(1): 107-119.
- [14] WESTFAL L. Software requirements engineering: what, why, who, when, and how [J]. **Software Quality Professional**, 2005, 7(4): 17-26.
- [15] BEN-MENACHEM M. Towards management of software as assets: a literature review with additional sources [J]. **Information and Software Technology**, 2008, 50(4): 241-258.
- [16] NIEMELÄ E, IMMONEN A. Capturing quality requirements of product family architecture [J]. **Information and Software Technology**, 2007, 49(11-12): 1107-1120.
- [17] XU L H, ZIV H, ALSPAUGH T A, et al. An architectural pattern for non-functional dependability requirements [J]. **Journal of Systems and Software**, 2006, 79(10): 1370-1378.
- [18] XIONG W, WATANABE Y, SHINDO H. A description approach to software by HOQ extension concept and its quantitative structuralization [J]. **Journal of Software**, 2005, 16(1): 8-16.
- [19] 熊伟. 质量机能展开 [M]. 北京: 化学工业出版社, 2005: 38-40.