

# 缺陷需求分析与模型

严玉清<sup>1,2</sup> 李师贤<sup>1</sup> 梅晓勇<sup>1</sup>

(中山大学计算机科学系 广州 510275)<sup>1</sup> (广东外语外贸大学信息学院 广州 510420)<sup>2</sup>

**摘要** 需求管理是软件过程改进的重要活动,需求缺陷管理是需求管理的重要组成部分,其目的在于检测、分析、解决和预防需求缺陷。从社会因素和技术因素两大方面分析了需求工程阶段缺陷产生的原因,对缺陷进行了分类。建立了需求(模型)缺陷列表、缺陷需求分析模型和基于需求缺陷管理的需求过程模型。指出需求缺陷列表所反映的需求分析过程的数量特征,能够为评估需求工程师能力、准确和全面地定义需求成熟度、研究需求演化波及效应、研究需求模型复杂度和需求缺陷分布规律提供必要数据和有益启示。

**关键词** 缺陷原因,缺陷类型,需求缺陷列表,需求演化

## Analysis of Defect Requirements and Management Model

YAN Yu-qing<sup>1,2</sup> LI Shi-xian<sup>1</sup> MEI Xiao-yong<sup>1</sup>

(Department of Computer Science, Zhongshan University, Guangzhou 510275, China)<sup>1</sup>

(School of Informatics, Guangdong University of Foreign Studies, Guangzhou 510420, China)<sup>2</sup>

**Abstract** Requirements management is an important activity for improving software process. Defect requirements management is a key component, and it involves detecting, analyzing, solving, and preventing defects. The causes of requirements defects were identified from two aspects, social and technical, and then classified. Three kinds of works were presented: defect lists of requirements and requirements models; the analytical modeling of requirements defects; and the modeling of the requirements process based on requirements defect management. Quantitative analysis of requirements defect lists can help in three areas: the evaluation of the performance of engineers; the correct definition of the mature degree of the requirements process; and the study of several components including the ripple effect, the complex degree of the requirements model, and the distributive rules of the requirements defect.

**Keywords** Cause of defect, Type of defect, List of requirements defect, Requirements evolution

## 1 引言

软件工程过程可看作是软件产品缺陷(defect)管理过程。软件产品在不同阶段其缺陷表现形式是不一样的。在需求工程过程中,即使不起眼的缺陷,若未察觉并加以纠正和修复,都将会随着开发活动的进行而形成滚雪球似的扩大蔓延,轻则造成纠正错误的代价急剧上升而损失巨大,重则致使项目返工或失败。调查表明,缺陷越早发现,修改缺陷所导致的错误所需的开销越小。文献[3]指出,在系统生命周期的需求阶段发现错误与在维护阶段发现错误的费用支出比为1:200。需求错误是项目中所发现的最大一类错误,占所发现错误的41%~56%,返工所需开销一般要占项目总费用的45%。因此在需求分析阶段及早发现缺陷及错误,建立良好的缺陷分析和管理机制,最大限度地使缺陷发生控制在软件开发的早期阶段具有重要现实意义。

## 2 需求缺陷的原因及其分类

软件需求是指用户对目标软件系统在功能、行为、性能、

设计约束等方面的期望。需求模型是相关需求的聚集,是在对应用问题及其环境的理解与分析基础上,为问题涉及的信息、功能及系统行为而建立的概念视图。需求工程过程的最终输出是需求规格说明及相应的需求文档,需求模型只是其中间产品。要保证获得高质量的需求规格说明,要从需求及需求模型入手。为了保证需求的正确性,首先要了解导致缺陷需求的原因并对其进行分类,这有利于缺陷的识别和预防。

### 2.1 需求缺陷原因

在需求分析阶段,把各种因素所导致的需求不完善看成需求缺陷,包括显式或隐式的错误(不论其大小)、需求变化所导致的需求冲突及不一致性等。

程序开发是一种社会和人类行为,是个人行为的有效组合。需求分析同样是人类行为组合的过程,是客户与掌握程序开发技术的人共同协作的结果。因此,我们从社会因素和技术因素两大方面来探讨需求缺陷的原因,这里排除那些人为的、恶意引入的缺陷。

#### 2.1.1 社会因素

软件开发涉及客户(投资人员、终端用户、软件市场营销

到稿日期:2008-05-13 本文受广东省自然科学基金项目(No. 06017089, 8151030007000002)资助。

严玉清(1963-),女,博士研究生,主要研究方向为软件工程、粗糙集理论, E-mail: yyqelsa@yahoo.com.cn; 李师贤(1944-),男,教授,博士生导师,主要研究方向为软件工程、形式语义学; 梅晓勇(1974-),男,博士研究生,副教授,主要研究方向为 Petri 网和面向服务计算。

人员等)及开发人员(软件工程师、项目经理等)。客户最初提出来的需求不清晰、不完整,需要需求分析人员根据自己有限的知识和开发经验,采用一定的软件工程开发范型和需求工程方法进行需求分析,并进一步挖掘客户需求。在需求获取与分析过程中,导致需求缺陷的一个重要原因是与人的因素有关,例如:(1)客户不知道自己想要什么;(2)请求客户来设计软件;(3)客户没有积极参与;(4)客户的期望值太大;(5)分析人员对应用系统领域知识了解不多;(6)没有确定合理的项目范围;(7)分析人员缺乏与客户的及时沟通;(8)客户和开发人员之间在项目成功的共同责任方面没有给予足够的重视。

另外,开发环境的动态性也是导致需求缺陷的一个重要原因。动态性的一个突出表现就是需求的变化(包括对需求模型进行的任何修改,如增加、删除、合并等)。变化会导致前后需求的不一致性(包括波及效应对其它需求所产生的不良影响)。关于需求的变化,其主要原因是:(1)中等以上规模的软件开发涉及大量需求,增加了需求分析的难度;(2)需求获取之初就存在未经过严格定义的需求;(3)法律、政策、市场、业务等发生了变化;(4)客户改变了主意;(5)开发人员未能很好地理解用户需求,或者遗漏客户感兴趣的需求,或者增加了自己的特殊理解,甚至自作主张修改了需求;(6)使用环境发生了变化;(7)出现了新的技术优势;(8)开发资金不足等。

这些因素会导致需求的不完备、需求遗漏、需求渗漏(需求蔓延)、需求冲突及不一致等。

### 2.1.2 技术因素

对于需求分析来说,所采用的需求工程技术(方法和工具)对需求分析的效率起到决定性的作用,它是需求分析能力的一个重要体现。从技术层面上看,影响需求分析效率有两大重要因素:(1)所采用的软件开发模型;(2)所采用的需求分析方法。

软件开发过程是需求演化的过程,是基于现有的知识和经验对未来软件系统进行预想的结果。为了获得良好的预想效果,需要建立开发过程模型,由方法按步骤对需求的内容和表现形式进行不断修正和设计。软件开发过程各个阶段之间的关系不是顺序的、线性的,而是带有客户和开发人员反馈意见的迭代过程。软件开发模型给出了软件开发活动各阶段之间的关系,它是软件开发过程的概括,为软件开发过程提供原则和方法。常见的软件开发模型有:瀑布模型、原型模型、螺旋模型、变换模型等。每种模型的开发指导思想都是正确的,但是其实践过程效率却是不一样的。例如:瀑布模型要求在软件开发的初始阶段即指明软件系统的全部需求,没有用演化观点看待这一过程,这是困难和不现实的;原型开发能够尽快向客户提供原型,使用户容易明确自己的真正所需,能统一客户和开发人员对软件项目需求的理解,有助于需求的定义和确认;螺旋模型能够强化客户和开发人员的风险意识,尽早避免开发进入僵局;变换模型能够提供形式化的开发方法,有效避免需求出错。在实践中通常会几种模型组合使用,以便充分利用各种模型的优点。

常见的需求分析方法是:面向数据流的方法(一种结构化的分析方法)、面向对象的方法、面向数据的方法(面向数据结构的系统开发方法 DSSD 及 Jackson 系统开发方法 JSD)、面向问题域的方法。这些方法同时也是需求建模方法,并且都

具备语言机制,供需求分析人员表达用户需求和构造软件模型。众所周知,这些方法的效率同样是各不相同的。需求分析方法的效率体现在:能够准确理解、正确表达、充分反映客户需求,能够采用尽可能先进、快捷的需求分析技术和工具(如需求复用、CASE 工具)以保证在一定的进度限制下出现少的需求缺陷。效率的体现还与需求分析人员的经验丰富程度、个性差异、理解能力、对需求分析方法的掌握及熟练程度等有关,也与项目特点有关,同时还要考虑需求分析的复杂度。在确保客户需求能够被充分挖掘、正确表达的前提下,衡量需求分析效率的一个简单方法就是低的需求缺陷率。降低需求缺陷率、提高需求模型的可测试性,最有效的方法就是采用形式化方法,采用形式化的需求表达语言,如 Z 语言、VDM 的 Meta-IV、CSP、Petri 语言等,采用形式化的开发工具,如基于数据流方法的需求分析 CASE 工具 DFA-Tool,它具有可视化形式化和可执行需求规格说明的动态分析能力。事实上,每种需求分析方法都可开发出其相应的形式化 CASE 分析工具。

从技术角度来看,导致需求缺陷的主要原因是:(1)没有有效综合多种软件工程范型;(2)未能对客户与软件开发人员进行开发前培训;(3)没有制定科学合理的需求分析进度计划表;(4)没有明确需求分析的目的;(5)没有或不注重使用需求分析工具及其它软件开发工具;(6)不注重需求复用技术;(7)没有建立正确的数据字典,以便保持术语的一致性;(8)没有采用有效的、适用的需求表达方法;(9)不注重收集、整理开发经验;(10)过多或不足的测试用例;(11)不完备的需求与测试用例的链接等。

## 2.2 缺陷需求分类

根据以上的分析,可把需求缺陷分为社会的和技术的,但为了更有针对性地识别、检测需求缺陷和错误,更好地挖掘客户需求,应该从缺陷的性质特征来对需求缺陷进行分类。

需求工程阶段的最终产品是带有详细说明文档的需求规格说明书,中间产品是需求模型。这一阶段产品质量定义与通常软件产品质量定义是不一样的。好的需求工程产品首先建立在好需求之上。好需求有以下一些标准:必要且正确的、可检验的、可达到的、非歧义的、完备的、一致的、可跟踪的、可分配的、简捷的、与实现无关的、标准结构的、惟一标识符的。违背了这些标准的需求便认为是缺陷的需求或错误需求。需求规格说明书具备的特征是:(1)正确性:需求规格说明书中的功能、行为、性能描述必须与用户对目标软件产品的期望相吻合;(2)无歧义性:对于用户、分析人员、设计人员和测试人员而言,需求规格说明书中的任何语法单位只能有惟一的语义解释;(3)完备性:需求规格说明书不能遗漏用户需求,即目标软件产品的所有功能、行为、性能约束以及它在所有可能情况下的预期行为,均应完整地包含在需求规格说明书中;(4)可验证性(可测试的):对于规格说明书中的任意需求,均应存在技术和经济上可行的手段进行验证和确认;(5)一致性:需求规格说明书的各部分之间不能相互冲突和相互矛盾。在术语使用方面、功能和行为特征方面以及时序方面要保持前后一致;(6)可理解性:需求规格说明书对于用户、设计人员和测试人员要具有可理解性。特别是对于非计算机专业的用户而言,不宜在说明书中使用太多的专业化词汇;(7)可修改性:需求规格说明书的格式和组织方式应保证能够比较容易地接纳

后续的增、删和修改,并使修改后的说明书能够较好地保持其他各项属性;(8)可跟踪性:需求规格说明书必须将分析后获得的每项需求与用户的原始需求项清晰地联系起来,并为后续开发和其他文档引用这些需求项提供便利。

需求变化是导致需求错误和缺陷的主要原因,会导致出现一系列不良的需求。根据好需求的标准,可把缺陷需求作以下分类:(1)标识符不规范或不惟一;(2)冗余的需求。指该需求不是必要的,是多余的;(3)非原子的需求。指该需求在功能上或性能上仍然可以进一步分解;(4)不一致的需求。指该需求与其它需求相互矛盾或发生分歧;(5)有二义性的或模糊的需求。指该需求有多种语义解释,或有些需求的表达缺乏可理解性,不知其所言;(6)位置错误的、时序错误的或优先级错误的需求。指没有明确某一需求在所有需求中的位置或所处的开发级别,也许把次要的需求作为了重要需求;(7)不正确的需求。指该需求没有被正确表达(如语法或语义不正确)或不符合用户需求;(8)不可验证的需求。

作为需求集,可把发生在需求模型或需求规约中的缺陷分类为:(1)冗余的:指需求模型内部或需求模型之间在功能或性能上有重叠或特征交互,可能导致冲突的发生或留下一个缺口;(2)不完备的或遗漏的:指开发人员没有充分理解和挖掘客户需求,导致客户需求遗漏或描述不完整;(3)不一致的:指由于需求的变化或其它原因导致需求模型内部或需求模型之间产生相互冲突、矛盾或分歧;(4)不正确的:指需求规格说明书中的功能、行为、性能描述没有与用户对目标软件产品的期望相吻合;(5)其他需求缺陷,例如开发人员缺乏开发经验和软件工程知识,从而缺乏预见性,未能正确地对需求的变化作前瞻性的预测而影响系统版本的升级;或者由于不正确的需求修复导致缺陷;或者需求测试用例之间会出现矛盾。

### 3 需求缺陷分析与管理模型

软件工程研究院 SEI 的软件过程能力成熟度模型 CMM 在软件行业有非常大的影响力,但其重点只在软件开发,不包含需求工程过程,因此文献[16]创建了一个类似的需求工程过程成熟度三级模型。其第二级——可重复级强调了需求管理问题。需求缺陷预防与需求跟踪和变更管理一样都是需求管理的重要活动内容<sup>[1]</sup>,要特别加以强调。有资料显示<sup>[3]</sup>,软件产品缺陷的 80%以上是在需求定义阶段引入的,如果能够提供的好的需求,则可以消除 80%的返工问题。所以,在需求工程阶段必须做好缺陷需求的识别、标记、纠错工作,充分预想将来可能的缺陷和错误,探索需求缺陷发生规律,逐步形成一套好的需求缺陷管理方法,达到防患未然。

#### 3.1 缺陷需求分析列表

文献[2]指出:“避免错误的引入是一种良好的工程习惯。设想错误将会发生并采取措去努力发现那些错误,无论如何是明智的”,“把性能需求汇集在文档的一个单独部分是有用的,这样做还有助于找出遗漏的性能需求”。也就是说,对每一类需求,以适当的易于检索的方式把其汇集在一起是明智和有用的。文献[9]指出一个良好的需求管理原则就是建立需求数据库,用列表(如:需求列表,需求跟踪列表,需求模型列表等)方式把所有采集到的需求各方面的信息记录下来(但有些信息无法用这种方式来表达,只能求助于纯文本方式)。这些表格应该是需求分析简单而有效的工具<sup>[12]</sup>。可利

用这种工具去检查相关的需求文档、计划等是否满足客户及开发需求,读者可以按照列表中罗列的项目,逐条检查,并建立起评估标准。有效使用这些工具是良好和成熟需求工程管理的体现,使得开发工作有据可依、有据可查,使得需求在整个开发过程中得以公开、公正的方式进行增加、删除、跟踪、查错和修改。

评价软件开发是否成功通常基于四个基本要素:(1)可容忍的进度延误;(2)可容忍的成本超支;(3)满足用户需求;(4)维护不困难。评价的主要目的在于如何帮助开发人员提高开发效率,并给开发人员提供有益反馈,总结开发工作的强项和弱项,对开发某类产品给出良好建议,对今后的进一步开发给出有益的预示、提醒作用。软件质量评价与一般产品的质量评价不同,不存在一个可以容忍的误差范围,一个看似很小的错误,却可能导致整个软件项目的失败,导致对整个开发过程的否定。所以,对于软件的评估重在找缺陷和错误,找弱项而不突出强项,尽管总结、评估开发过程的强项也是一项有意义的工作。

软件开发效率评估遵循多元质量评估机制,需求分析效率评估也应该建立在多元评估体系之下:(1)客户参与开发的能力,对开发工作的支持和关注程度;(2)需求分析师的知识背景及开发经验;(3)需求分析师与客户关系是否密切;(4)需求分析师采用了什么新的技术、方法和工具;(5)需求工程阶段需求及缺陷管理是否形成制度,能力如何等。

需求缺陷管理机制要与需求管理机制相兼容,即对需求缺陷的跟踪要与对需求的跟踪和需求变更的跟踪和控制相一致。因此,简单而有效的需求缺陷分析与跟踪方法是建立缺陷需求列表,如同需求(跟踪)列表一样。为了体现简洁性,为了达到易于操作及有效性,表格不能太复杂,不同表格之间具有某种可比性、联想性,这些表格要能够提供一定的数据去对需求缺陷进行多种数据分析(如统计分析)。

在需求管理中,需求跟踪表是很重要的一种列表。按照 ANSI/IEEE 给出的标准,关于需求跟踪,有以下两项要求:(1)每项需求的来源都很清晰;(2)在未来的开发或改进文档中便于引用。更具体一点,就是需要记录以下需求信息<sup>[3]</sup>:(1)需求的来源;(2)该需求存在的理由;(3)有哪些需求与该需求有关;(4)在系统设计、实现和用户文档中如何满足该需求。

另外,需求模型列表也是一种重要的列表,它可以包含以下属性:模型标号、模型组成(由需求标号组成)、版本号、模型描述语言、作者、重要度、实现层次及难度、构建日期、是否被修改。

尽管需求管理贯穿在从需求获取之初到软件交付使用整个生命周期全过程,但从有效管理的角度来看,需求管理应该分阶段进行。对于需求缺陷,不同阶段有其不同的内涵和表现形式,所以一个好的做法是分阶段、有区别地对缺陷进行检测、分析、跟踪和解决。

需求列表和需求模型列表应该分成两种,一个是针对客户,一个是针对开发人员。针对客户的需求(模型)列表记录需求当前的最新版本和状态,每条需求的内容用自然语言来表达,以方便用户理解,得以及时与用户沟通。在针对开发人员的需求(模型)列表中,需求(模型)可用形式化语言或专业开发术语来表达,除了反映需求(模型)的最新版本和状态之

外,也要反映设计阶段与编码阶段的相关信息<sup>[7]</sup>,更重要的是要与需求(模型)缺陷列表相链接,可用需求(模型)标号作为链接的关键字。另外,一定要做好历史信息的记录与保存工作,这是良好需求管理工作的一个见证。表1列出了可针对开发人员的需求列表所包含的属性(亦可根据实情情况作增删)。

表1 需求列表

需求标号	需求内容	需求来源	基本理由	事件用例编号	位置	需求类型	相关的需求	优先级	稳定性	版本	是否已被实现	作者	日期
------	------	------	------	--------	----	------	-------	-----	-----	----	--------	----	----

其中,需求标号可采用层次化编码方法<sup>[11]</sup>进行惟一标识;需求来源应该记录;是新采集的需求,或从别的需求中分解出来的,或由其它需求合并的,或对某一需求修改得到的;基本理由是指为什么要描述该需求,即描述该需求的原因,它可以记录:该需求可能是项目参与者的需要、机构质量标准、规范、工作陈述、技术文档、事件报告或其他需求。例如:系统需求需要第101号需求,以便使系统的用户可以收到其请求

被发送的反馈;事件用例编号;需要这项需求的事件与用例清单;可按几种不同方式对需求进行分类<sup>[13]</sup>,例如:(1)按需求内容分为功能、性能、运行需求;(2)按用户的期望分为正常、期望、兴奋需求;(3)按需求的必要性分为强制、希望、任选需求。还可使用多重方法对需求进行分类<sup>[3]</sup>;需求优先级有几种,例如:五等级——关键,后续关键,后续重要,改良,可选;三层次——基本/有条件/可选。这有助于项目参与各方确定系统的核心需求,有助于设计人员确定系统体系结构以及解决设计冲突。

表2、表3分别列出了需求缺陷列表及需求模型缺陷列表所包含的内容。其中,缺陷原因分为:社会的、技术的;缺陷类型按2.2节的分类进行;对缺陷需求进行分级,例如:“暂缓”,“这个版本不解决,下一个版本再解决”等;修改方法指对该需求(模型)作了何种修改:增加、删除、分解、合并、重写(对需求本身进行修改,由不正确变为正确),每种修改方式都会导致需求(模型)的变化。对于以上数据库(表格),根据需要可以对它们进行分类、排序、链接、自动更新、统计等与数据库相关的操作。

表2 需求缺陷列表

需求标号	需求内容	版本号	所属模型标号	缺陷原因	需求类型	缺陷类别	缺陷描述	发现者	发现缺陷用时	发现日期	修改难度	缺陷级别	修改方法	修改后的内容	修改日期	被修改的次数	修改者	修改累积次数	付出的工作量
------	------	-----	--------	------	------	------	------	-----	--------	------	------	------	------	--------	------	--------	-----	--------	--------

表3 需求模型缺陷列表

需求模型标号	模型组成	版本号	模型类型	模型重要度	缺陷类别	缺陷描述	发现者	发现日期	发现缺陷用时	修改难度	缺陷级别	修改方法	修改后模型组成	修改日期	修改者	修改累积次数	付出的工作量
--------	------	-----	------	-------	------	------	-----	------	--------	------	------	------	---------	------	-----	--------	--------

在需求分析时,要保留需求中所有修改的信息以及修改前的所有历史记录,并将修改进程记录在系统维护日志中。

### 3.2 缺陷需求(模型)列表所反映的数量特征

著名计算机科学家、丹麦学者 Dines Bjørner 指出<sup>[8]</sup>:“在整个软件开发过程的每一阶段和步骤中,我们不再有任何理由不使用形式化技术了。”对此,我们的理解是:不仅在需求的描述中需要使用形式化技术,而且在开发过程和需求过程中需要使用数学方法去揭示过程的数量特征。

表1、表2和表3可以为定量研究需求分析过程提供必要的信息。例如:(1)为评估需求工程师的能力提供数据。可以统计某一需求分析人员正确分析、建立需求(模型)的数量,查找缺陷的数量及所花的时间,对缺陷进行正确修改的数量等,为评估和计算需求工程师的能力及所付出的工作量提供依据;(2)为更加准确和全面地定义与需求过程成熟度有关的指标提供数据。文献<sup>[14]</sup>基于需求演化角度,从一个实际工业案例(avionics industrial safety-critical case)中获得了22个不同软件版本的需求变化统计数据,建立了需求成熟度、平均需求变化数、需求稳定度、历史需求成熟度等数量指标去加深人们对需求演化过程的理解。这一工作对揭示需求演化及需求过程本质提供了有益的指导作用,但不够全面和准确,仍然需要进行更深入的数据收集工作;(3)为研究需求演化波及效应及计算需求模型复杂度积累数据。波及效应可被定义为对某些需求的修改影响到其它需求的现象。这种效应是由于在需求(模型)之间存在着各种各样相互联系引起的。通过需求(模型)缺陷列表所记录的信息,为有效建立需求演化波及

效应数量化框架及计算需求模型复杂度提供必要数据;(4)为研究需求缺陷分布规律提供数据。通过统计某一时刻或时间范围内,某一版本或某些版本的缺陷需求数量,为研究需求缺陷分布规律提供可能。

关于以上这四方面的工作,我们将另文研究。

### 3.3 缺陷需求分析与需求过程模型

#### 3.3.1 缺陷需求分析模型

图1表示缺陷需求分析模型。当检查出缺陷需求后,仍然要根据需求列表提供的信息对相关需求进行检查,以避免因波及效应导致不良后果。

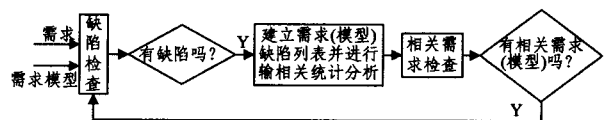


图1 缺陷需求分析模型

#### 3.3.2 基于缺陷管理的需求过程模型

文献<sup>[3]</sup>指出:需求工程的进展相当缓慢,系统和软件开发工作一直受到与需求有关的问题的困扰,因此要特别注重需求工程过程。文献<sup>[15,16]</sup>指出,把需求错误的检测、分析、解决、预防尽可能控制在需求工程阶段是良好的软件开发习惯,并且为了获得稳定的、持续的需求质量,必须对需求过程进行控制。

需求工程的目的是建立待解决问题的概念框架,把定义不足的问题转化为定义良好的问题以便找出好的解决方案。在把定义不足的问题转化为定义良好问题的过程中,需求的

缺陷管理起到了关键的作用。

图2表示以缺陷管理为主的需求过程模型。

从图中可见,需求工程过程是一个多重叠代的过程,需求管理无处不在,缺陷管理是其中一个重要内容。不同开发阶段需求管理的内容不完全一样,最为关键的是要在需求工程过程中强化需求管理意识,建立有针对性的需求管理原则和规范。

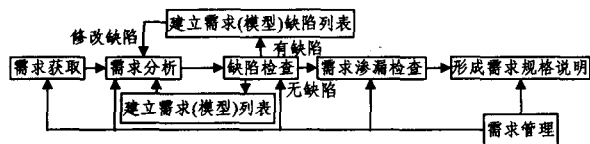


图2 以缺陷管理为主的需求过程模型

## 4 需求缺陷与需求演化研究的联系与区别

### 4.1 联系

仅从字面上来看会让我们有一种错觉,认为需求演化是导致需求缺陷的原因。我们在文献[6]中分析了大量的需求演化研究文献,从中可见,需求缺陷与需求演化研究本质上是一样的,都是对由于多种原因(特别是需求的变化)所导致的不完善需求及需求之间的分歧、矛盾(统称为需求不一致性)进行检测、分析、解决,并建立预防机制(如跟踪机制)。由Nuseieh, Hunter, Steve Easterbrook等所建立的基于观察点(Viewpoints)的不一致性管理框架尽管是基于观察点的需求分析展开,而且主要研究由矛盾需求所产生的不一致性,但纵观其整个框架可以看出,只要把不一致性的内涵扩大,把指导这一框架实施的各个步骤的原则作适当的修改<sup>[6]</sup>,就完全可把这一框架变为需求缺陷管理模型。

### 4.2 区别

尽管关于需求缺陷与需求演化在研究内容上本质是一样的,但从目前的研究方法和手段来看,它们仍然存在一些区别:(1)关于需求缺陷研究更多地是从加强需求跟踪、改善需求过程的角度出发<sup>[1,3]</sup>(需求管理是这一改进过程的重要活动),而需求演化则自成体系建立需求演化模型和不一致性管理框架<sup>[6]</sup>;(2)需求缺陷研究手段单一,仅进行定性研究,不注重收集缺陷特征数据从而进行定量研究;或者仅用一些简单统计数据去说明减少需求缺陷(错误)的必要性<sup>[3]</sup>,缺乏系统性、科学性与合理性;未能从理论上揭示缺陷发生发展规律,未能从实践中获取大量数据再借助数学手段去揭示这种规律。相比之下,需求演化研究工作要起色得多。不仅从管理的角度进行定性研究,也从抽象角度建立演化模型,利用形式化方法定义各种需求不一致性,用多种逻辑方法和数学方法去分析、推理需求不一致性,建立对需求不一致性的跟踪、检测机制,建立各种解决不一致性的方案。但由于缺乏演化数据收集,因此对需求演化的研究更多停留在定性层面,缺乏定量研究模型。

### 4.3 建议

软件工程理论研究者和实践者应该真正关注需求缺陷与需求管理研究,把需求演化并入需求缺陷研究范畴,用系统化的、工程的、管理的方法辅以逻辑的、数学的以及其他学科的知识 and 手段加强需求缺陷管理,既要区分需求缺陷在开发不同阶段的不同表现,也要用系统化的、整体论的观点去观察需求缺

陷的在整个开发过程的综合表现。从社会(特别是人)和技术角度揭示缺陷发生、发展的规律,真正建立起有效的缺陷预防机制,达到防患于未然,并建立合理的软件生命周期质量观。

**结束语** 需求管理是软件生命周期的重要活动内容,需求缺陷管理是需求管理的重要组成部分,是提高软件开发生产率、揭示软件开发规律、正确定义软件质量的一项基础性工作。目前,关于软件缺陷、软件质量的研究更多是集中在代码编写阶段,未能真正系统地、全面地去纵览整个生命周期由于需求缺陷所导致的质量问题。因为不同开发阶段需求的表现形式不一样,需求缺陷的表现形式也不一样,要获得全局的质量观,首先要从局部做起、从各个不同阶段做起、从对生命周期不同观察做起。需求工程是软件生命周期的重要和第一阶段,关于这一阶段的需求缺陷研究已有不少成果(包括需求演化的研究)<sup>[1,3,6,14]</sup>。这些成果对需求缺陷研究起到里程碑作用,对需求缺陷的进一步研究有很好的启发和指导作用。但是,从已有这些研究成果还未能真正捕获需求过程整体质量观,不能揭示开发人员的开发素质和工作效率的关系,不能揭示缺陷发生、发展的数量特征和规律,也未能建立有效的缺陷预警和预防机制。鉴于此,我们想在这些方面做一些工作,根据需求缺陷列表建立一些反映需求分析过程的数学模型,从定量角度去研究需求分析过程,以期揭示需求分析过程的一些本质现象和规律。

## 参考文献

- [1] (美)Younessi H. 面向对象的软件缺陷管理[M]. 赵文耘,沈斌,等译. 北京:机械工业出版社,2004
- [2] (英)K. Bray I. 需求工程导引[M]. 舒忠梅,罗文村,李卫华,等译. 北京:人民邮电出版社,2003
- [3] (美)Young R R. 有效需求实践[M]. 韩柯民,耿民,等译. 北京:机械工业出版社,中信出版社,2002
- [4] (美)杰拉尔德·温伯格. 程序开发心理学[M]. 邓俊辉,译. 北京:清华大学出版社,2003
- [5] 齐治昌,谭庆平,宁洪. 软件工程[M]. 北京:高等教育出版社,2004
- [6] 严玉清,李师贤. 需求演化方法[J]. 展望
- [7] (英)Robertson S, Robertson J. 掌握需求过程(第2版)[M]. 王海鹏,译. 北京:人民邮电出版社,2007
- [8] Björner D. Software Engineering 3: Domains, Requirements, and Software Design. Beijing: Tsinghua University Press, 2007, 8
- [9] Kar P, Bailey M. Characteristics Of Good Requirement // International Committee on Systems Engineering (INCOSE) Symposium. 1996
- [10] 刘寅斌. 系统分析之路[M]. 北京:电子工业出版社,2005
- [11] 覃征,何坚,高洪江,等. 软件工程与管理[M]. 北京:清华大学出版社,2005
- [12] (美)Hoberman S. 数据建模——分析与设计的工具和技术[M]. 贾爱霞,程耀,译. 北京:机械工业出版社,2004
- [13] 张虹主编,姜淑娟,刘迎春,荣新节编著. 软件工程与软件开发工具[M]. 北京:清华大学出版社,2004
- [14] Anderson S, Felici M. Quantitative aspects of requirements evolution // Proceedings of the 26th Annual International Computer Software and Application Conference. COMPSAC, Oxford, England, IEEE Computer Society, 2002; 27-32
- [15] (英)杰拉尔德·温伯格. 质量. 软件. 管理:一阶测量[M]. 李先华,邢彦,张红,译. 北京:清华大学出版社,2005
- [16] (英)Sommerville I, Sawyer P. 需求工程[M]. 赵文耘,叶恩,等译. 北京:机械工业出版社,中信出版社,2003, 8: 177-206