

基于业务场景的用例粒度划分范式

赵玉明¹, 舒红平^{1,2}, 魏培阳^{1,2}, 刘魁^{1,2}

(1. 成都信息工程大学软件工程学院;
2. 成都信息工程大学软件自动生成与智能信息服务重点实验室, 四川 成都 610225)

摘要: 用例驱动整个统一软件开发过程, 但用例划分缺乏统一标准规范, 从而导致用例划分不够准确。针对该问题, 以业务场景为基础, 对用例粒度划分展开研究, 提出采用3种范式规范用例粒度划分。从用例划分源头、建模阶段及实际工程规模展开进行分析, 3种范式为建模人员在具体业务场景下的用例划分提出解决方案, 可为建模人员节省建模时间、提升建模效率, 从而完善系统架构。

关键词: RUP; 业务场景; 软件建模; 用例划分; 范式

DOI: 10.11907/rjdk.182425

中图分类号: TP302

文献标识码: A

文章编号: 1672-7800(2019)006-0052-04

The Normal Form of Use Case Granularity Partition Based on Business Scenario

ZHAO Yu-ming¹, SHU Hong-ping², WEI Pei-yang^{1,2}, LIU Kui^{1,2}

(1. School of Software Engineering, Chengdu University of Information and Technology; 2. Key Laboratory of Software Automatic Generation and Intelligent Information Service Chengdu University of Information and Technology, Chengdu 610225, China)

Abstract: The use case drives the whole software rational unified process(RUP). However, the division of use cases lacks unified standards and norms, which leads to inaccurate division of use cases. In view of this problem, three norm forms are proposed to standardize the partition of use case granularity. In this paper, the use case granularity partitioning is studied based on business scenarios, and three norm forms are proposed to standardize the use case granularity partitioning in the modeling phase. The source of use case partitioning, the stage of modeling and the scale of the actual project are analyzed and the three norm forms provide solutions for use case partitioning in specific business scenarios. Through the three norm forms, modelers can save modeling time, improve modeling efficiency and the system architecture.

Key Words: RUP; business scenario; software modeling; use case granularity; norm forms

0 引言

在软件工程领域, RUP既是一套软件方法, 又是一种典型的软件开发模式。它以迭代增量式架构为中心, 用例驱动的软件开发方法为主要特征, 其中用例驱动是贯穿软件开发始终的方法。用例驱动整个软件架构的建立, 软件架构是逻辑层次较高的系统视图, 是系统规划与设计的高

层次抽象^[1]。

但是, 在用例粒度划分上, 没有一套完整的规范, 导致对于同一个业务场景, 不同的建模人员划分出来的用例可能呈现不同程度的差异。同时, 大多数建模人员并没有相关领域知识, 在建模过程中更多依靠建模理论知识及自身业务经验。由于缺少具体业务场景与相关规范, 无法准确挖掘业务中的用例。

美国学者 Wiegiers 突出说明了用例在软件需求工程中

收稿日期: 2018-10-31

基金项目: 四川省科技厅科技支撑项目(18ZDYF3256); 四川省教育厅科研项目(18ZB0126)

作者简介: 赵玉明(1993-), 男, 成都信息工程大学软件工程学院硕士研究生, 研究方向为大数据背景下的软件自动生成与智能服务、面向领域的软件工程方法与技术、机器学习; 舒红平(1974-), 男, 博士, 成都信息工程大学软件工程学院教授、软件自动生成与智能信息服务重点实验室主任, 研究方向为软件开发环境与智能服务、面向领域的软件工程方法与技术、领域大数据; 魏培阳(1984-), 男, 硕士, 成都信息工程大学讲师、软件自动生成与智能信息服务重点实验室研究员, 研究方向为软件自动化生成、软件需求设计一体化等; 刘魁(1983-), 男, 硕士, 成都信息工程大学讲师、软件自动生成与智能信息服务重点实验室研究员, 研究方向为深度学习、领域大数据等。

的重要性,提出针对具体的业务场景,用例应该有粒度,同时还提出用例与软件架构之间的关系,但是并没有对用例划分提出统一指导意见,也没有指出如何从具体业务场景中推导出用例。文献[3]在提出用例重要性的同时,发现一个系统用例个数需保证在 20 个左右,但是没有提出相关理论依据,也没有结合实际工程规模。文献[1]虽然给出了用例划分方法,也提出应根据建模阶段、工程规模及实际业务场景划分粒度,但更多的是基于工程实例,没有形成完整的理论规范。

基于以上分析,本文一方面结合国际前沿需求工程知识,另一方面,结合国内研究与工程实践,以文献[6]~[9]的工程实践为依托,提出从一个具体业务场景出发,从用例实质入手,以建模阶段及实际工程为依据,整合出 3 种用例划分范式,使划分出来的用例向上可以追溯业务场景,为软件架构设计提供基础,向下可以推导需求,为验证需求提供依据,从而节约建模时间、提升建模效率。

1 影响用例粒度划分的 3 个因素

用例粒度划分存在 3 个方面的问题,也是影响用例划分 3 个重要因素。

(1)在具体业务场景下,业务边界模糊不清^[5]。由于缺少边界,不同的建模人员对业务进行抽象提炼的方向不同,包含在边界内的用例也会呈现出不同程度的紊乱。

(2)混淆建模阶段^[4]。建模可以分为不同的阶段,如果没有对软件建模所处的阶段有一个清晰的认识,划分出来的用例粒度很难适合业务实际需要。

(3)缺乏对实际工程量的宏观把握。有的软件规模相对较大,有的较小。在软件建模时,需要对软件实际规模进行宏观把控,提高划分准确度与目的性。

2 用例粒度划分范式设计原理

根据以上分析,本文结合用例实质与具体应用场景,提出用例粒度划分的 3 种范式。

第一范式:主要以建模过程中用例粒度划分的实质为基础。如果要使划分的用例正确,首先需要有一个清晰的业务边界^[6],边界大小直接决定建模者对业务的抽象层次。所以第一范式有效约束了划分用例的源头,以免用例粒度划分在开始阶段出现错误。

第二范式:主要为在具体业务场景下,关于建模阶段的分析。在业务建模阶段,建模人员应该关注具体业务,建立一个业务用例模型,解决实际业务问题。而在系统建模阶段,用例的确定通常从业务用例模型中推导而来,这样才能保证每一个系统用例均来自于实际业务。

第三范式:解决实际工程量的问题。有的工程量很大,可能需分解才能完成,这样在实际用例选择上也需依据实际情况而定。

3 3 种范式具体实现过程

3.1 第一范式

在该范式(1NF)中,根据用例粒度划分的本质,提供划分方法。

(1)确立边界。边界可以帮助建模者弄清楚现阶段处于哪个抽象层次,如果没有边界或者边界混乱,仅凭建模人员主观判断,往往导致混乱,无法准确获取实际需求,划分的用例粒度合理性也难以得到保证。

(2)确立涉众期望。在确定好一个边界后(假设这个边界是正确的),首先需找到边界内的涉众(Stakeholder,与系统所有相关的对象,指在此边界内部的对象,有可能非人),然后,整理涉众对系统的价值诉求,即涉众期望。

将每个期望设为一个集合,首先,确保边界内任意两个集合之间的交集为空,即用例之间不能存在内容交叉、重复等情况;然后,集合总数保持在 20 个左右,以便把握模型复杂度^[2]。该过程可以用以下数学方法分析:

设该边界内包含的用例全集为 U ,涉众的某个期望为一个子集,子集设为 $E_i(i=1\cdots n)$,则 U 应当满足公式:

$$E(1) \cap E(2) \cap \cdots \cap E(n-1) \cap E(n) = \varnothing \quad (1)$$

$$0 \leq \sum_{i=1}^n E(i) \leq 20 \quad (2)$$

最后,取出集合元素,形成相应的用例。

过程如图 1 所示。

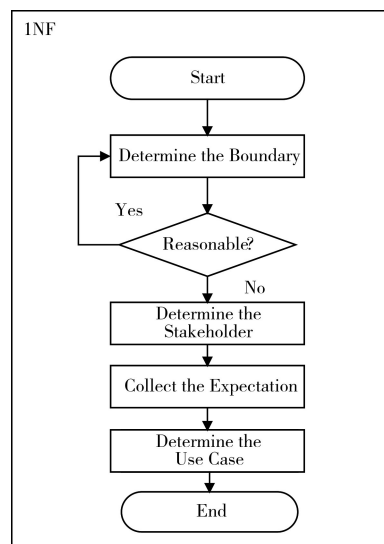


图 1 第一范式分析过程

通过该流程图,模式系统一算法步骤如下:

算法:1NF 算法

输入:合理的边界

输出:合理的用例粒度

系统输入一个边界 Boundary

While Boundary 不合理 do

 选择合理的边界

End

确定涉众 S

For S(i)

收集涉众的期望,形成集合,满足数学公式

确定用例

3.2 第二范式

在第二范式(2NF)中,一方面,可以很好地解决不同建模阶段用例粒度划分出现的错误;另一方面,可以对第一范式的用例进行优化。

第二范式(2NF)设计如下:

(1)分清建模阶段,根据所处阶段划分用例粒度。

(2)在需求分析阶段,使用 UML 建模时,主要分为业务建模、概念建模与系统建模。概念建模是业务建模与系统建模之间的过渡阶段,本文主要探讨业务建模与系统建模的用例粒度把握问题。在业务建模阶段,用例需能够描述一个完整的业务流程,以便确定需求范围。

(3)在系统建模阶段,该用例需能描述与计算机的一次交互。

因为用例是一个独立的个体,用例划分后需保证用例之间不存在高度耦合、内容重复等问题。合理的用例粒度划分格式如图 2 所示,不合理的用例划分如图 3 所示。

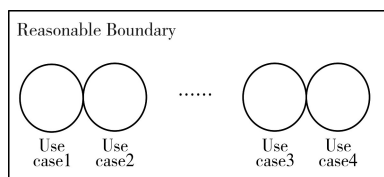


图 2 合理的用例划分格式

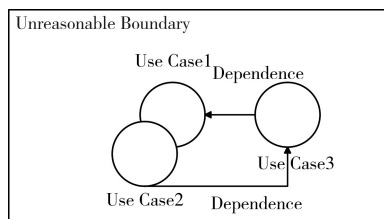


图 3 不合理的用例划分格式

第二范式过程如图 4 所示。

第二范式算法步骤如下:

输入:确定建模阶段

输出:合理的用例粒度

建模所处的阶段 Stage

If Stage=业务建模阶段

While 与一个业务流程无关 do

选择一个能说明一个业务流程的用例

Else

While 与计算机的交互无关 do

选择一个可以与计算机相互的用例

End

3.3 第三范式

第一范式与第二范式是在理想情况下分析、总结出来的,并没有考虑实际工程规模的问题。第三范式(3NF)首

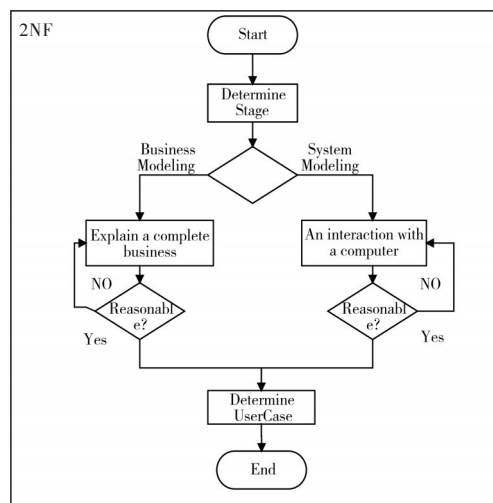


图 4 第二范式分析过程

先考虑工程量问题,然后划分工程参与人数、时限等,即工作包问题,并根据工作包确定工程规模。最后,按照工程规模划分用例粒度。

在工程量非常大时,一般要选择大的用例。相反,如果选择小的用例,则对于大的项目可能产生几百个用例,与第一范式冲突,并且每个员工实际工程量也非常大,导致数量过大、过于细碎而无法控制。如果从宏观上进行把握,采用较大的用例粒度,将有助于控制需求范围,减少需求遗漏。在工程量较小时,一般选用较小的用例,如果此时选择较大的用例,则会使采集到的需求过于模糊而容易忽略细节,过程如图 5 所示。

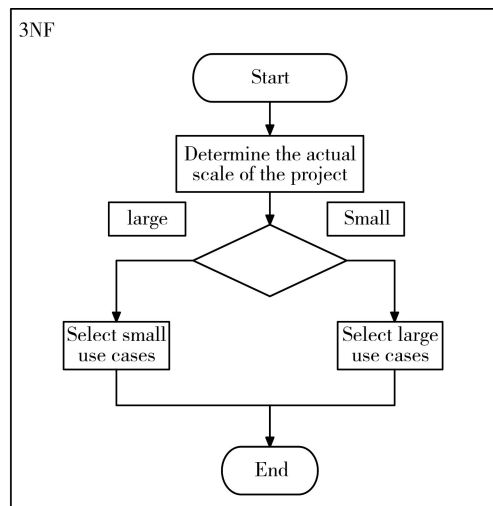


图 5 第三范式分析过程

模式系统三的算法步骤如下:

输入:确定工程的规模

输出:合理的用例粒度

准确判断工程规模 Project scale

If Project scale 非常大

按照模式系统一、二的要求选取用例粒度

Else

按照模式系统一、二的标准来选取用例粒度
End

4 案例分析

4.1 案例选择

该案例是一个网上购物系统,主要涉及买家、卖家及系统管理人员。在建模过程中,首先按照传统方式建模;然后,基于业务场景采用 3 种范式进行建模。通过将两种用例划分方法对比,可以发现在第二种情况下用例划分更加合理,可为后续软件架构建设提供根本保障。

4.2 过程分析

4.2.1 传统建模

按照传统建模方式,建模人员首先整理涉众期望,进行分析和处理;然后整理出对应用例,画出具体用例图。根据该方式画出的用例如图 6 所示。

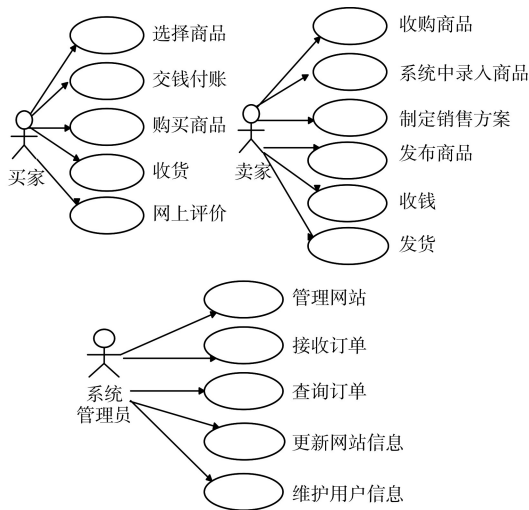


图 6 业务用例

从图 6 可以看出,用例个数已达到 16 个,并且用例大小不一,没有具体业务场景作为支撑。有的甚至是一个步骤,使架构上两个模块具有非常紧密的联系,有强依赖关系的逻辑被分配到架构上要求独立的模块。

4.2.2 采用 3 种范式的建模

以业务场景为基础,结合 3 种范式,展开对用例粒度的划分。

首先结合第 3 范式判断工程规模,可以看出其项目相对较小,用例选择方法应相对详细,符合整体划分原则。

根据第一范式,确定业务边界。经过分析,只有卖家与买家对系统有非常明确的期望,而系统管理员是被动接收买家与卖家在网上的业务。所以根据具体业务场景,结合实际情况,业务边界结构图 7 所示。

买家主要目的与期望是购买商品,而卖家是卖商品。系统管理员通过维护网站从而为用户服务。选择物品、录入商品信息等皆为买卖商品的一个步骤,所以基于该业务场景及边界,用例粒度划分如图 8 所示。

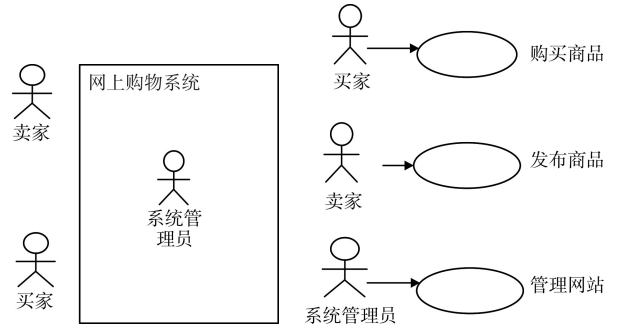


图 7 卖家边界结构

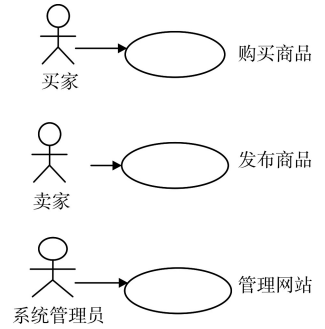


图 8 买家边界结构

然后根据第二范式,明确具体的建模阶段,对以上业务用例粒度再进行划分。在系统阶段,首先需保证分析来源是业务用例,然后保证系统用例适合在计算系统中执行,最后保证划分的用例可追溯到具体业务场景。图 9 是结合买家购买商品的系统用例划分。

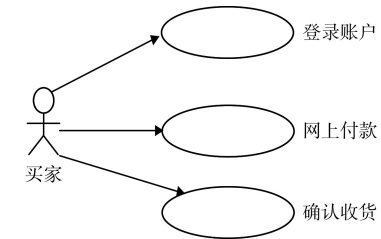


图 9 系统建模阶段买家用例划分

4.2.3 对比分析

传统划分方法缺乏相应业务场景,粒度大小不一,有的甚至把步骤作为用例对待,难以调控、修改,很容易混淆建模阶段,无法为软件架构设计提供合适的用例,也没有为认识业务系统提供更高的抽象层次,用例复杂度相对较高。

与传统划分方法相比,采用 3 种范式方法可以根据具体业务场景,以业务主角目标为依据进行划分用例,严格避免将步骤当作用例,还可以根据具体业务边界进行用例调控及修改;结合建模阶段划分用例,使划分的用例为以后设计开发提供很好的基础;同时划分的用例简单明了、层层递进,向下可推导需求,向上可追溯业务场景,也为架构实现提供依据。

5 结语

针对建模阶段用例粒度划分不准确的问题,本文从实际业务场景出发,在确定的业务场景下根据用例本质、建模阶段及实际工程规模,提炼出用例粒度划分的 3 种范式。通过实际案例验证发现,3 种范式为用例粒度划分提供了方向、方法与规范,避免了用例粒度划分不准确的问题。同时通过 3 种范式划分出来的用例,向上可以追溯场景、为软件架构设计提供方向,向下可以映射需求,提升需求把控的灵活性。

参考文献:

[1] 谭云杰. 大象-Thinking in UML[M]. 北京:中国水利水电出版社,

(下转第 59 页)

均值分别为:0.039、0.059 和 0.044。结合表 5 可知,ItemCF-emoji 算法 F1 值的 Top1~5 平均值比 itemCF 算法提高了 0.023。

表 5 各算法 F1 值的 Top1~5 平均值

	ItemCF-emoji	itemCF
F1 均值	0.044	0.021

4 结语

emoji 表情已成为互联网时代必不可少的元素之一,是人们表达情感的重要载体。通过分析海量 emoji 表情信息,可以了解用户情绪状况、对某个社会现象的观点,以及对某款产品的偏好等。本文从用户情感角度入手,将 emoji 表情引入到 ItemCF 推荐算法中,但推荐效果依赖于评论中 emoji 表情转化为用户情感的准确性。因此,未来工作可对文中 emoji 表情进行更加细致的分类,使提炼出的用户情感更为精确,从而提高推荐效果。

参考文献:

[1] 刘建国,周涛,郭强,等.个性化推荐系统评价方法综述[J].复杂系统与复杂性科学,2009,6(3):1-10.

[2] 周军锋,汤显,郭景峰.一种优化的协同过滤推荐算法[J].计算机研究与发展,2004,41(10):1842-1847.

[3] 邓晓懿,金淳,韩庆平,等.基于情境聚类 and 用户评级的协同过滤推荐模型[J].系统工程理论与实践,2013,33(11):2945-2953.

[4] 熊聪聪,邓滢,史艳翠,等.基于协同过滤的美食推荐算法[J].计算机应用研究,2017,34(7):1985-1988.

[5] 邓爱林,左子叶,朱扬勇.基于项目聚类协同过滤推荐算法[J].小型微型计算机系统,2004,25(9):1665-1670.

[6] 罗辛,欧阳元新,熊璋,等.通过相似度支持度优化基于 K 近邻的

协同过滤算法[J].计算机学报,2010,33(8):1437-1445.

[7] 荣辉桂,火生旭,胡春华,等.基于用户相似度的协同过滤推荐算法[J].通信学报,2014,13(2):16-24.

[8] 马宏伟,张光卫,李鹏.协同过滤推荐算法综述[J].小型微型计算机系统,2009,30(7):1282-1288.

[9] 赵屹星.Emoji 表情中的受众心理研究[J].设计艺术研究,2016,6(1):46-49.

[10] 金芊芊.Emoji 表情符号的艺术价值[J].常州工学院学报:社科版,2017,35(3):52-55.

[11] 杨莞桐.青年群体 emoji 表情符号使用情况调查与研究[J].人间,2016,217(22):17-19.

[12] TANG D, QIN B, LIU T. Document modeling with gated recurrent neural network for sentiment classification[C]. Conference on Empirical Methods in Natural Language Processing, 2015: 1422-1432.

[13] NOVAK P K, SMAILOVIĆ J, SLUBAN B, et al. Sentiment of emojis[J]. PloS one, 2015, 10(12): e0144296.

[14] VIDAL L, ARES G, JAEGER S R. Use of emoticon and emoji in tweets for food-related emotional expression[J]. Food Quality & Preference, 2015, 49(6): 119-128.

[15] 许雅.移动社交语境下 Emoji 表情的传播态势与功能探究[J].新媒体研究,2016,2(8):24-28.

[16] JÄSCHKE R, MARINHO L, HOTH O A, et al. Tag recommendations in folksonomies[M]. Knowledge Discovery in Databases: PKDD 2007. Springer Berlin Heidelberg, 2007:506-514.

[17] BATAGELJ V, ZAVERŠNIK M. Generalized cores[J]. arXiv preprint cs/0202039, 2002.

[18] KRIPPENDORFF K. Agreement and information in the reliability of coding[J]. Communication Methods and Measures, 2011, 5(2): 93-112.

[19] HAYES A F. SPSS macro for computing Krippendorff's alpha[J]. Retrieved September, 2006, 6(4):17-20.

[20] 项亮.推荐系统实践[M].北京:人民邮电出版社,2012.

(责任编辑:黄健)

(上接第 55 页)

2011.

[2] WIEGERS K E. Software requirements[M]. 刘伟琴,刘洪涛,译.北京:清华大学出版社,2004.

[3] SOUZA V, MYLOPOULOS J. From awareness requirements to adaptive systems: a control-theoretic approach[C]. Proceedings of the 2nd Int'l Workshop on Requirements@Run.Time, 2011:1-7.

[4] GOMAA H. Software modeling & design UML, use cases, pattern, & software architecture[M]. 北京:机械工业出版社,2016.

[5] 栗元邦,彭蓉,季晶晶,等.经验研究中情景感知需求获取与建模系统文献综述[J].软件学报,2018,29(2):320-329.

[6] 杨芙清.软件工程技术发展思索[J].软件学报,2005,16(1):1-7.

[7] 邹盛荣.UML 面向对象需求分析与建模教程[M].北京:科学出版社,2016.

[8] 杨长春.实战需求分析[M].北京:清华大学出版社,2016.

[9] 张家重,徐家福.需求工程研究新进展[J].计算机研究与发展,1998(9):1-5.

[10] 陶传奇,李必信, Jerry GAO,等.基于模型的构件软件修改影响分析[J].软件学报,2013,35(1):942-960.

[11] 张莉,蒲梦媛,刘奕君,等.对软件工程中经验研究的调查[J].软件学报,2018,29(5):1422-1450.

[12] 赵玮.面向对象软件工程中软件需求分析[J].山西师范大学学报:自然科学版,2016,20(2):26-28.

[13] 王聪,王智学,徐友云.基于 UML 的面向 C4ISR 能力需求分析的对象建模语言[J].2015,42(2):150-156.

[14] 董威,舒绍娟,徐小平.软件需求工程课程建设思考与实践[J].计算机工程与科学,2014,36(A2):34-27.

[15] 王瑞雪,张涛.UML 模型驱动的划分测试用例生成方法研究[J].计算机应用研究,2012,29(9):3334-3337.

[16] 潘加宇.用例粒度么[J].程序员,2008(3):72-74.

[17] 钱雪忠,宋建生.基于 UML 图和不同粒度切片的回归测试研究[J].计算机工程与科学,2012,34(11):124-129.

[18] 李玉琴,赵文耘.从领域需求到产品线体系结构的映射——一种面向特征的方法[J].计算机研究与发展,2007,44(7):1236-1242.

[19] 万江平,安诗芳,黄德毅.软件工程知识体系指南综述[J].计算机应用研究,2016,23(10):1-3.

[20] 王继成,高珍.软件需求分析的研究[J].计算机工程与设计,2002,23(8):18-21.

(责任编辑:江艳)