

敏捷需求过程建模

张国生

(云南大学 信息学院, 云南 昆明 650091)

摘 要: 将敏捷方法融入软件需求工程过程的每一个活动之中, 充分发挥敏捷方法和计划驱动方法的优势, 体现了规范的力量和安慰以及敏捷的宽松和创造性, 并用双变迁 Petri 网 DTPN 为敏捷需求工程过程活动建立反馈、迭代模型。

关键词: 计划驱动方法; 敏捷方法; 双变迁 Petri 网; 需求过程活动; 反馈; 迭代

中图分类号: TP311

文献标识码: A

文章编号: 1000-7180(2012)02-0027-04

Agile Requirements Processes Modeling

ZHANG Guo-sheng

(School of Information, Yunnan University, Kunming 650091, China)

Abstract: In this paper, agile method is embedded into software requirements engineering processes activities. The methodology brings sufficiently into play advantages of agile method and plan-driven method, embodies strengths and comforts of disciplines and looseness and creativities of agility, builds feedback and iterative model for agile requirements engineering processes activities with DTPN.

Key words: plan-driven method; agile method; DTPN; requirements processes activities; feedback; iteration

1 引言

在软件系统的生命周期中, 其过程也是不断变化和逐步完善的^[1]. 敏捷方法已经发展成为一种新的软件开发方法, 这些方法被成功地应用在一些中小型项目中, 敏捷方法对聚焦简化和改进软件开发过程提供了实在的软件开发方法, 敏捷开发方法避免了软件错误的集中爆发^[2]. 近几年, 敏捷方法常常作为一种针对传统开发方法的成功选择, 特别是当应用环境为快速需求变化和短周期发布时^[3]. 软件组织常常需要对市场动态、新用户需求和技术更新做出反应, 许多关于敏捷方法的实践和调查表明: 快速开发改进了敏捷团队的交流、合作及其团队、用户和业务单元, 许多组织将敏捷方法看作是软件开发中表述关键问题的一种方法, 敏捷方法如: 极限编程、SCRUM 等已经被建议作为对变化、缩短开发时间、改进交流与合作, 特别是将时间作为一个组织的关键竞争优势时的方法^[4].

需求过程是具有识别、分析、建档和确认系统需求的传统软件工程过程, 需求工程和敏捷方法常常被认为是不相容的, 需求工程主要依赖于共享知识文档, 而敏捷方法关注用户与开发者之间面对面的合作. 需求工程关注需求的识别、建模、交流和建档, 在需求工程过程中有许多有效的技术可以保证需求的完整性、一致性和相关性, 需求工程的目的是在系统开始之前帮助理解构建什么并防止高昂的重做, 这个目标是建立在如下两个假设之上的: (1) 错误发现得越晚其修改的代价越昂贵; (2) 系统设计和实现之前确定稳定的需求集是可能的.

2 传统方法和敏捷方法比较

2.1 传统方法

传统方法即计划驱动的方法, 包括瀑布模型、统一过程和 V 模型等, Hirsch^[5]指出: 所有的计划驱动方法都具有如下特征: 软件需要的功能/属性被事先规约; 从项目的开始到结束被构建了详细的计划;

收稿日期: 2011-05-08; 修回日期: 2011-07-13

基金项目: 国家自然科学基金项目(60463002); 云南大学重点教改项目(WX070142)

需求被进一步规约,然后被严格的变化请求过程执行;实现开始之前体系结构和设计规约必须完成;编程工作只集中在程序阶段;项目完成后进行测试;质量保证用形式化方法处理.传统的方法侧重于计划和架构,解决环境变更的手段是在开发中依靠预见力,有了这种预见力,架构师常常能在变更对系统造成副作用前将其消灭于无形中,这种计划驱动方法关注的是软件制品的质量和过程的可预见性.软件的系统工程观点认为,在通过从需求到已完成的代码等一系列代表物来推动软件开发的过程中,计划驱动方法非常精确地依赖于明确的步骤,过程中每个步骤文档的完备性至关重要,这是在此基础上对每个代表制品进行彻底验证的重要保证,早期,常常将软件开发周期看作是从概念直到最终产品的一个瀑布,最近,增量和演化过程的观点已被采用,但对于需求、设计和编码,仍然有很强的编档和可跟踪性要求.软件作为一种产品,其可塑性使其有必要支持过程中产品定义的多重版本,过程的定义和管理是计划驱动方法的关键,正因为如此,计划驱动方法几乎总是和过程改进联系在一起,过程需要进行定义、标准化并需要逐步改进以提供控制、管理其操作所需的数据,这种过程通常包含有详细的计划、活动、工作流程、角色和职责以及工作产品描述.

2.2 敏捷方法

由于小的迭代和用户交互,有助于更好地推动交流和反馈,交流的好处在于传递知识,敏捷方法进一步建议有现场客户,开发人员可以感觉到频繁反馈的价值,现场客户可以控制整个过程和项目,开发进展可以定期反馈给客户,从工作环境的角度,敏捷项目被认为是涉众之间尊重、信任和保证质量的宽松方法;敏捷方法轻量化了过程,鼓励开发人员摆脱重型过程的束缚,拥抱变化,投入敏捷的怀抱;敏捷方法主张短的交付周期,客户的紧密参与,并在思想上倾向于适应变更而不是预见变更.敏捷方法提出了敏捷宣言:个体和交互胜过过程和工具;可以工作的软件胜过面面俱到的文档;客户合作胜过合同谈判;响应变更胜过遵循计划.敏捷方法是一种非常轻量的过程,它应用短的迭代周期、主动使客户参与到需求的建立、优先级设定和验证的工作中,它依赖于团队内部的隐式知识而非文档,一个真正的敏捷方法必须包含的所有属性有:迭代(数个周期)、增量(不是一次就交付整个产品)、自组织(由团队来决定处理工作的最好方法)以及自然浮现(过程、原则、工作结构是在项目进行过程中形成的,不是预先确定的).

对于成功的项目,敏捷和规范缺一不可,软件系统正变得越来越大、越来越复杂,每一次成功的冒险,既需要敏捷又需要规范,如果只有强用力的规范而缺乏敏捷,终将导致官僚作风,进而停滞不前,而缺乏规范的敏捷则如同一个新创建的公司盈利之前所表现出的那种不负责任的狂热,优秀的公司会根据自身的目标和所处的环境对这两者做出恰当的权衡.

本文将敏捷方法融入软件需求工程过程的各个活动之中,在需求过程的每一个活动中引入敏捷方法,一方面借助于传统方法的规范性,另一方面借助于敏捷方法的自由性,使得软件系统需求更加规范、完备、快捷,为系统的进一步设计、编码、测试、交付、维护奠定良好的基础.

3 敏捷需求规范和敏捷需求工程过程

3.1 敏捷需求规范

需求工程是所有需求处理活动的总和,它收集信息、分析问题、整合观点、记录需求并验证其正确性,最终反映软件被应用后与其环境互动形成的期望效应.Wiegers^[6]提出优秀需求所具有的特性为:完整性、正确性、精确性、可行性、必要性、无歧义和可验证.

根据优秀需求的特性,本文提出如下敏捷需求规范:

- (1) 短周期和连续、频繁地发布有价值的轻量需求规格说明文档;
- (2) 欢迎客户变更需求;
- (3) 在整个需求过程生命周期中,不断迭代、增量、反馈客户需求变更;
- (4) 充分依靠现场业务人员;
- (5) 涉众之间面对面交流是最高效和有效的方法;
- (6) 简化、重构、反省、演化、持续改进需求设计,快速满足客户需求;
- (7) 利用先进、有效、高效的技术;
- (8) 团队自组织,充分调动团队成员积极性;
- (9) 过程、原则、工作结构自然浮现;
- (10) 反复审查、验证需求规格说明文档是否正确和完整地表达了客户对系统的需求.

3.2 敏捷需求工程过程

需求开发的主要活动包括:需求获取、需求分析、需求规格说明、需求验证.将敏捷需求规范充分融入需求开发的每一个活动中,使传统方法和敏捷

方法有机结合,既发挥了计划驱动方法严格、可靠、规范的优势,又发挥了敏捷方法交流、简化、反馈、勇气的价值。

需求获取是从客户、文档或者环境中获取需求的过程。该过程是涉众之间面对面交流最多的时期,敏捷方法强调专职、工作在一起的客户代表,需求获取的成功依赖于具有如下品质的客户代表:易于协作(Collaborative)、有代表性(Representative)、有授权(Authorized)、尽责(Committed)和在行(Knowledgeable),即 CRACK。将 CRACK 型客户融入开发过程中,不管是传统方法还是敏捷方法都是十分重要和必要的,直接和定期地与这些客户交互,并将客户需求作为需求过程反馈、迭代、反省、增量、重构、演化的重要依据,是项目成功的关键要素。

需求分析主要通过建模来整合各种信息,从而使人们更好地理解问题,同时,需求分析还会为问题定义一个需求集合,这个集合能够为问题界定一个有效的解决方案,需求分析还需要检查需求中存在的错误、遗漏、不一致等各种缺陷,并加以修正。利用先进、有效、高效的技术对获取的客户需求进行分析,分析方法、过程、原则、工作结构由团队自行组织,充分调动团队成员积极性,最终获得必要、完备、一致、可靠的分析结果,当需求存在冲突时,通过涉众之间的优先级协商解决,有争议的需求被确定为关键需求。

获取的需求被编写成文档,其中项目前景和范围文档记录业务需求,客户需求文档记录客户需求,需求文档便于涉众之间的交流,优秀的需求文档是无二义、完备、正确、可理解、一致、精确和可靠的。在敏捷方法中,需求文档的编写采用轻量方法,根据获取的需求快速编写需求文档,然后,与客户进行面对面交流,再修改、完善需求文档,最后,得出客户满意的需求规格说明。

为了尽量不给设计、实现、测试等后继开发活动带来不必要的影响,需求规格说明文档中定义的需求必须正确、准确地反映客户的意图,需求验证用于实现这样的功能。在需求工程过程中,客户需求为客户向开发人员提出的素材(Stories),敏捷方法通过与客户面对面交流获取的客户素材验证需求规格说明文档,并反馈、增量、演化修改需求规格说明,直到需求规格说明能尽可能准确、一致地反映客户需求为止。

要得到尽可能准确、一致地反映客户需求的需求规格文档,整个需求过程的活动需要不断地反馈、

增量、迭代、演化,需求过程的每一个活动都与敏捷方法密切相关,构成完整的需求过程生命周期,每一个需求过程活动也需要借助于敏捷方法不断地反馈、增量、迭代、演化,因此,每一个需求过程活动也构成一个子周期,如图1所示。

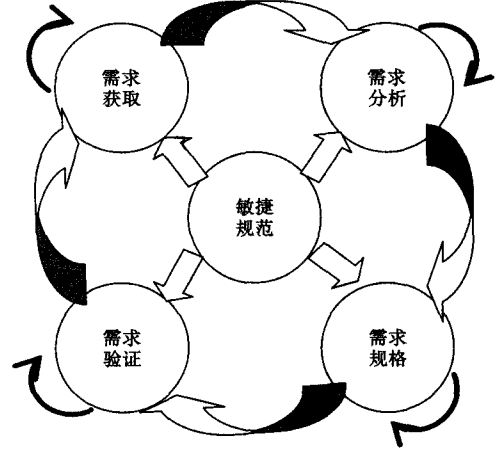


图1 敏捷需求过程的迭代反馈周期

4 敏捷需求过程形式化方法

4.1 双变迁 Petri 网 DTPN

根据敏捷方法不断迭代、增量、反馈客户需求变更,简化、重构、反省、演化、持续改进需求设计,快速满足客户需求的规范下,引入双变迁 Petri 网 DTPN^[7]进行形式化建模。DTPN 在原型 Petri 网的基础上,将变迁扩充为两类:数据变迁和反馈控制变迁;弧也扩充为两类:与数据流相关的弧和与反馈控制流相关的弧。在 DTPN 中包含三种类型的节点:库所 P 、数据变迁 T 、反馈控制变迁 Q 。以下给出 DTPN 的形式化定义:

定义1 DTPN 定义为一个8元组

$$DTPN = (P, T, Q, F_D, F_C, W, G, M)$$

其中, $P = \{p_1, p_2, \dots, p_n\}$ 为有限库所集; $T = \{t_1, t_2, \dots, t_m\}$ 为有限数据变迁集; $Q = \{q_1, q_2, \dots, q_h\}$ 为有限反馈控制变迁集; $F_D \subseteq (P \times T) \cup (T \times P) \cup (P \times Q) \cup (Q \times P)$ 为数据流弧集; $F_C \subseteq (P \times Q) \cup (Q \times P) \cup (P \times T) \cup (T \times P)$ 为反馈控制流弧集; $W: \{F_D \cup F_C\} \rightarrow \mathbf{Z}$ 为权函数, $W_D \subseteq (P \times T) \cup (T \times P) \rightarrow \mathbf{Z}^+$ (正整数集), $W_C \subseteq (Q \times T) \cup (T \times Q) \rightarrow \{\mathbf{Z}^+ \cup \mathbf{Z}^-\}$ (非0整数集), $W_D \cap W_C = \emptyset$, 且 $P \cap Q = \emptyset, P \cap T = \emptyset, T \cap Q = \emptyset$; G 是 Guard 函数, 是从库所到布尔空间的一个映射, 对变迁数据域的前集中的每个库所都有一个前提条件与之相对应, 定义为 $G: 2^P \rightarrow \{0, 1\}$; M 是标志函数, 是从库所集合 P 到 $\mathbf{Z}$ 的一个映射, 即

$M: P \rightarrow Z, M_0$ 为系统的初始标志向量。

定义 2 变迁前集和后集沿用原型 Petri 网的符号,即数据变迁 T 的前集用 $\cdot t$ 表示,后集用 $t \cdot$ 表示;反馈控制变迁 Q 的前集用 $\cdot q$ 表示,后集用 $q \cdot$ 表示。

定义 3 双变迁 Petri 网 DTPN 中变迁使能规则:

(1) 对于数据变迁 $T, \forall t_i \in T$, 如果其前集 $\cdot t_i$ 中至少有 $W_D(p_i, t_i)$ 个托肯, 并且其 Guard 函数 G 对于 $\cdot q_i$ 是有效的, 则变迁 t_i 可以发生, 即

$\forall t_i \in T, \forall p_i \in \cdot t_i, \forall p_j \in \cdot q_i: (M_D(p_i) \geq W_D(p_i, t_i)) \wedge (G(M_C(p_j)) = 1)$ 时, t_i 变迁可以发生;

(2) 对于反馈控制变迁 $Q, \forall q_i \in Q$, 反馈控制变迁 q_i 可以发生当且仅当在其前集 $\cdot q_i$ 中至少有 $W_C(p_i, q_i)$ 个托肯, 或者如果 $\cdot q_i = \Phi$, 则变迁 q_i 总是可以发生, 即

$\forall q_i \in Q, \exists p_i \in \cdot q_i: (M_C(p_i) \geq W_C(p_i, q_i)) \vee (\cdot q_i = \Phi)$ 时, q_i 变迁可以发生。

定义 4 数据变迁 t_i 将 M_k 变成 M_{k+1} 遵循如下规则:

(1) $\forall p_i \in \cdot t_i \Rightarrow M_{k+1}(p_i) = M_k(p_i) - W_D(p_i, t_i)$

(2) $\forall p_i \in t_i \cdot \Rightarrow M_{k+1}(p_i) = M_k(p_i) + W_D(t_i, p_i)$

定义 5 反馈控制变迁 q_i 将 M_k 变成 M_{k+1} 遵循如下规则:

(1) $\forall p_i \in \cdot q_i \Rightarrow M_{k+1}(p_i) = M_k(p_i) - W_C(p_i, q_i)$

(2) $\forall p_i \in q_i \cdot \Rightarrow M_{k+1}(p_i) = M_k(p_i) + W_C(q_i, p_i)$

4.2 敏捷需求过程模型

用 DTPN 既可以模拟敏捷需求过程中的反馈、迭代, 又可以模拟需求过程规范, 因此, 利用 DTPN 为敏捷需求过程建模, 在模型中, 用 T 变迁表示需求过程活动, 用 Q 变迁表示与客户进行面对面交流后的反馈, 并将客户需求不断地迭代入不同的需求活动中, 建立相应的敏捷需求过程模型如图 2 所示。

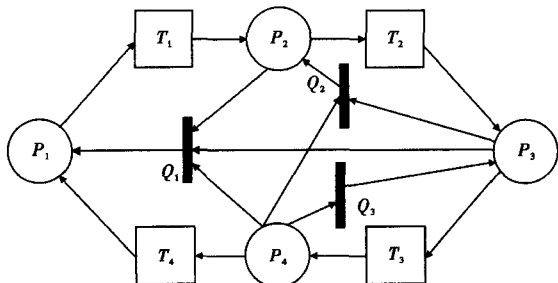


图 2 敏捷需求过程模型

图中, T_1 表示需求获取、 T_2 表示需求分析、 T_3 表示需求规格说明、 T_4 表示需求验证; Q_1, Q_2, Q_3 表示反馈控制变迁。

在敏捷需求过程模型中, T_1, T_2, T_3, T_4 四个主要需求活动构成了需求过程生命周期, 每一个需求活动都可以获得来自其他活动的反馈控制, 体现了敏捷方法的反馈、迭代、增量、快速发布等核心思想。

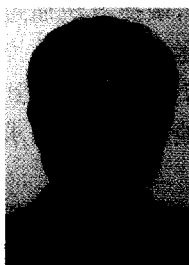
5 结束语

将敏捷方法融入需求工程过程, 可以很好地适应当前需求工程中, 客户需求不断变化, 而需求开发人员应尽可能快速地响应这种变化, 并对系统进行反馈、增量、迭代、演化, 使需求规格说明文档充分满足客户需求, 减少设计、实现、维护、发布等后续开发的错误。将敏捷方法融入到需求过程的每一个活动中, 将规范与简化、快速、灵活、勇气有机结合, 充分发挥传统方法和敏捷方法的优势, 不失为需求工程的良好方法, 有较高的应用价值。

参考文献:

- [1] 葛君伟, 葛智, 方义秋. 非线性动力系统在软件需求分析中的研究[J]. 微电子学与计算机, 2010, 27(6): 85-88.
- [2] 王世安. 基于敏捷开发的构件化软件可靠性研究[J]. 微电子学与计算机, 2011, 28(3): 119-122.
- [3] Nanda C. Using an ethnographic process to conduct requirements analysis for agile systems development[J]. Inf Technol Manage, 2008(9): 55-69.
- [4] Boehm B, Turner R. Balancing agility and discipline: a guide for the perplexed[M]. Boston: Addison-Wesley, 2004.
- [5] Hirsch M. Moving from a plan driven culture to agile development[C]//Proceedings of the 27th international conference on software engineering (ICSE 2005). USA: Missorui, 2005.
- [6] Wiegers K. Software requirements[M]. 2nd ed. New York: Microsoft Press, 2003.
- [7] 罗怡桂, 顾君忠. 一种新型 Petri Net: 双层变迁定时 Petri Net[J]. 计算机工程, 2002, 28(8): 45-47.

作者简介:



张国立 男, (1968—), 硕士, 副教授。
研究方向为软件工程及其形式化。