

# 基于敏捷化统一过程需求建模的研究及实践

段隆振<sup>1</sup> 王凤斌<sup>2</sup> 甘晟科<sup>2</sup> 黄水源<sup>1</sup>

(南昌大学信息工程学院 南昌 330000)<sup>1</sup> (南昌航空工业学院 南昌 330000)<sup>2</sup>

**摘要** 敏捷建模(AM)是适应需求变化、提高软件开发效率的新型建模思想,而 RUP 是 Rational 公司提出的一个通用的软件开发过程框架。本文分析了当前 RUP 实践的缺点及敏捷建模实践特点,在此基础上总结了敏捷化统一过程(AM-RUP),并提出了一种 AM-RUP 的需求建模过程——“总体概要需求获取,按需求优先级进行计划,小增量迭代式简单需求建模”。以社会保险系统的开发为背景,探索了基于 AM-RUP 的需求建模过程及技术。

**关键词** 软件工程, RUP, 敏捷建模, 需求建模

## The Research and Application of AM-RUP Requirement Modeling

DUAN Long-Zheng<sup>1</sup> WANG Feng-Bin<sup>2</sup> GAN Sheng-Ke<sup>2</sup> HUANG Shui-Yuan<sup>1</sup>

(College of Information Engineering, Nanchang University, Nanchang 330000)<sup>1</sup> (Nanchang Institute of Aerial Technology, Nanchang 330000)<sup>2</sup>

**Abstract** Agile modeling is a new modeling through which is able to adapt to the changes in requirement and improve the efficiency in the software developing. RUP is a general frame of the software development. This paper analyses the questions in the RUP practices and the features in AM practices. Based on this, the paper concludes AM-RUP, and brings forward the AM-RUP Requirement Modeling Process—"Getting general requirements, planning by requirement priorities, simple requirement modeling at the small increment and iterative mode". Based on the development of the social insurance system this paper explores the AM-RUP requirement modeling process and technology.

**Keywords** Software engineering, RUP, Agile modeling, Requirement modeling

当前,软件开发现状不容乐观,不少系统延迟、超支交付,甚至无法交付。为了解决软件开发面临的困难,学者们提出了众多的软件开发过程,其中 Rational 统一开发过程(RUP)是其中最具代表性的过程之一,也是被广泛采用的一种软件开发过程。同时,敏捷建模(AM)是当今最具突破性的建模思想,彻底摆脱了传统建模思想的束缚,但它本身并不是一种完整的方法论,需要结合到具体的开发过程中,Rational 统一开发过程良好的可裁剪性使得二者的结合成为可能。而需求建模的效率是软件开发过程中其它阶段工作的前提,是提高软件开发效率和质量的关键。将敏捷建模应用到 Rational 统一开发过程之中进行软件需求建模是很值得探讨的问题。

由于社会保险系统需求复杂性和易变性,本开发团队选择了基于敏捷化统一过程的需求建模过程和技术进行系统的需求获取分析,很好地简化了需求建模工作、提高了需求变更的速度及软件开发质量。

### 1 Rational 统一过程敏捷化

敏捷建模是一组软件建模阶段的指导性原则,是针对基于软件系统的有效建模和编写文档的一个混乱而有序的、基于实践的方法<sup>[1]</sup>,是 Scott W. Ambler 在众多敏捷方法学的基础上发展起来的实践的抽象和总结。敏捷建模思想视软件为首要目标,支持后续工作为第二目标。它主张轻装前进,创建简单内容、有目的的建模,通过快速反馈及递增变化达到包容需求变化的效果。有了这些敏捷建模原则,那要如何应用于实践呢?以下总结敏捷建模思想在软件项目开发的 4 类核心实践,并与当前统一过程的实践进行比较分析。

• 迭代增量简单建模(1)使用合适的制品:每一种制品各有优势和劣势,为了达到不同的目的,需要采用不同的制品,

以获得“物尽其用”的效果。而当前 RUP 实践主要使用 UML,然而 UML 并不能满足所有的使用需求,比如用户界面需求建模。(2)并行创建多个模型和迭代到其它制品:任何模型都不能满足全部建模的需要,对同一业务用不同模型获取不同侧面的需求。(3)小增量建模:较大的工作量分成能够随时间一个个发布的几个较小部分,最好是跨几个星期或一两个月左右的增量。该实践抵制了当前 RUP 实践的预先大型设计(简称 BDUF)思想,同时防止了花长时间进行建模及写文档的弊端。(4)用简单的工具创建简单的模型,该实践扩充了当前 RUP 实践使用的工具和模型。

• 团队有效协作:协同建模、公开展示模型、用户积极参与,以便快速达成共同愿景及建立良好的系统架构。

• 验证建模工作:由于敏捷开发主张短交付周期,编写代码来验证模型是最佳的方法。而当前 RUP 实践习惯预先大型建模,然后进行大量评审工作及相应的修改模型工作,接着才进行编码工作。

• 敏捷文档实践:丢弃临时模型、有危害时才更新模型、契约模型规范化。而当前 RUP 实践习惯将各种模型和文档及时更新以保持完全一致(即有很强的跟踪习惯)。

RUP 是一个以用例驱动、体系架构为核心、迭代递增式的过程,并广泛应用于各类软件的开发当中。由于 RUP 提供了极为丰富的内容,当前软件界常视其为复杂的过程,但常常忽略它的关键特点——可裁剪性。经过以上 AM 实践同当前 RUP 实践的分析,发现当前 RUP 实践所存在的主要问题在敏捷建模实践中均得到解决。因此,将 AM 融入 RUP 实践中能改善软件开发中超时、超支及低质量的现状。

综上所述,敏捷化统一过程(AM-RUP)是一个以用户为中心、需求驱动、小增量迭代、易变更的有序软件开发过程。

段隆振 教授,硕导,研究方向:软件工程、分布式计算、数据挖掘;王凤斌 助教,硕士,研究方向:数据库应用技术;甘晟科 助教,硕士;黄水源 讲师,硕士。

## 2 AM-RUP 需求建模

据 AM-RUP 的核心思想,即以用户为中心、需求驱动、小增量迭代、易变更的有序开发过程,本文提出“总体概要需求获取,按需求优先级进行计划,小增量迭代式简单需求建模”的 AM-RUP 需求建模方案。该方案对应的 AM-RUP 需求建模过程如图 1 所示,充分体现了分组并行、小增量迭代、多反馈、快变更的特点。

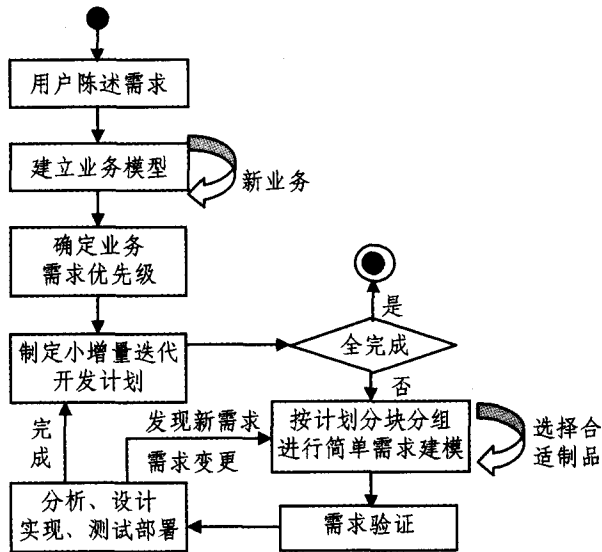


图 1 AM-RUP 需求建模过程

该需求建模过程基本依据 RUP 的工作流顺序及迭代的思想,而与当前 RUP 需求建模不同的关键在于“制定小增量迭代计划”。这使每一迭代周期中建模任务和内容得以缩小,并有利于 AM 的使用,进而原来繁琐的需求变更工作也将得以简化,最终提高软件开发效率。

## 3 社会保险系统的需求建模实例分析

### 3.1 社保系统的业务建模——总体概要需求获取

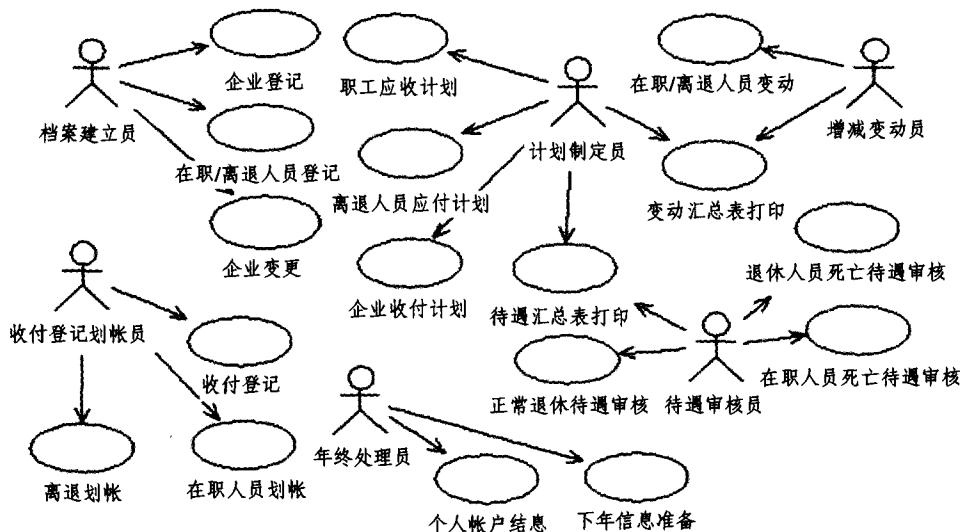


图 3 社会保险基本用例组图

#### 3.1.2 基本用例建模

上小节将整个社保业务分成了几个业务模块,为了理解这几个业务模块的高层需求,决定遵照 AM-RUP 提出的“有目的建模”的实践,同时创建几个基本用例模型<sup>[1]</sup>,如图 3 所

#### 3.1.1 社保业务流程

秉着 AM-RUP 的用户方积极参与的实践原则,本开发团队邀请了用户方一起建模——提供业务全过程图,这将防止理解的不正确并减少由此带来的反复向用户方求证的时间。当然,该过程图与业务流程图有些不同,但足以团队成员所理解。由于业务流程对社保中各种业务的逻辑关系进行了描述,有利于系统的分块和确定业务优先级,因此该模型是值得创建及保存的。为了更清楚地表示业务的逻辑性,将稍微改动的简单流程图来表示社保系统的业务流程,如图 2 所示,该图并没有详细地标出判断信息,但对于社保的业务已清晰,这遵循了 AM 的“创建简单的内容”的实践及“轻装前进”的原则。

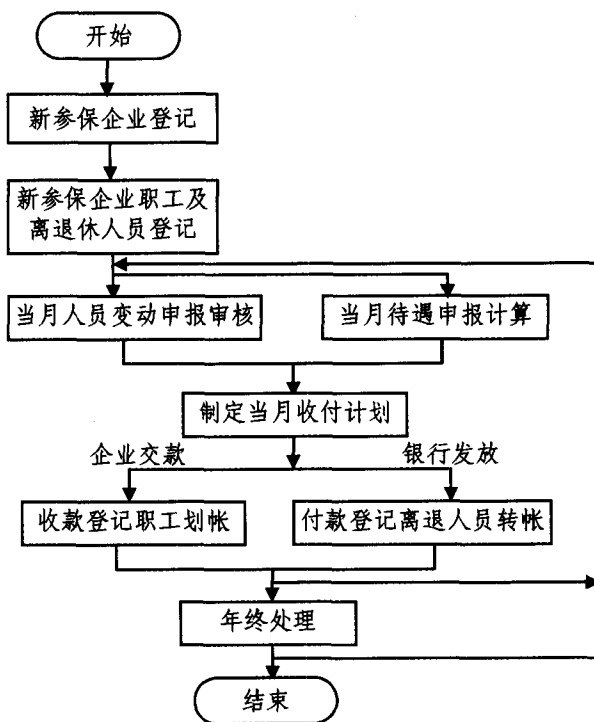


图 2 社会保险业务流程图

示。

#### 3.1.3 业务对象建模

为了与用户一起探索、讨论用来完成业务用例的业务实体,决定创建业务对象模型。在业务建模会议中,利用索引卡

及笔两种最普通的工具建立 CRC 模型(每张索引卡承载一个 CRC 模型),这样可以方便增添或更改 CRC 中的内容(图 4)。而且,每张 CRC 卡均可灵活地移动于各业务模块用例的业务实体讨论中,毕竟各业务模块间逻辑关系紧密,含有不少相同业务对象。

档案建立业务员	职工	企业	离退人员
企业登记	姓名、企业	企业名称	姓名、企业
在职人员登记	参保年月	职工人数	参加工作年月
离退人员登记	当前缴费基数	缴费基数	退休昔年人金额
企业变更	视同缴费年限	离退人数	退休时企业划金额
业务员号	离退实际缴费年限	离退休费类型	企业划余额
业务员名	人员状态	企业类型	状态

图 4 社保系统的部分 CRC 卡

以上 CRC 卡主要描述了档案建立业务模块中的几个业务实体——档案建立业务员、职工、离退休人员和企业的责任(即所需的数据及实体的行为)、同实体协助完成责任的其它实体。而对其它实体,具体是如何协助它完成责任,此阶段则可以不考虑,将在需求建模阶段中详细分析。至此,通过与社保业务人员的讨论和建模,已对社保系统的业务领域有了较清楚的认识。

### 3.1.4 编写补充业务规格说明书

针对将来可能出现的需求,依据 AM-RUP 建议的“包容变化”、“轻装前进”两原则,开发团队不必预先进行任何建模工作,以防止预先大型设计(BDUF)所带来的开发效率低的危害。因此,选择变例这一简单制品对社保系统潜在的需求进行描述(如表 1),并将其放入补充业务规格说明书中。

表 1 社保系统变例

新增:权限管理。
可能性:极大。
时间:一年后。
影响:以后的业务职责分配可能改变

该变例中的内容是制定社保项目开发迭代计划的依据之一。

## 3.2 确定业务优先级及制定初步小增量迭代计划——按需求优先级进行计划

### 3.2.1 快速处理突然的需求新增

在讨论业务优先级时,社保方又提出了一个专为个人参保(即以个人名义而不是以企业名义参保的业务)所用的系统。为了更好地理解,立即初步建立了个人参保的业务用例模型(图 5)。

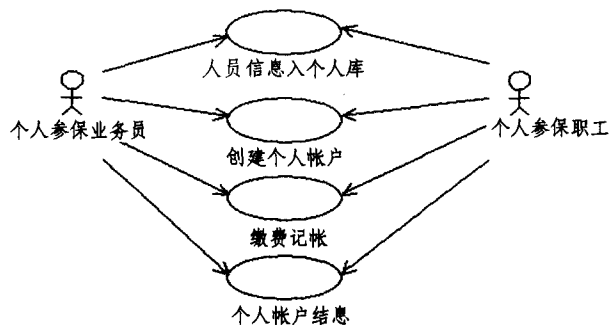


图 5 个人参保业务用例图

在处理突然新增需求的过程中,采取了积极面对新需求的态度,灵活应用简单的模型与用户进行交流,并对新需求的认识快速达成初步的一致。在此充分运用了 AM-RUP“创建简单模型”及“快速反馈”的实践。

### 3.2.2 确定优先级/制定小增量迭代计划

确定优先级及初步制定小增量迭代计划是 AM-RUP 需求建模过程中的关键。小增量迭代计划制定后,在一个开发周期内,各组开发人员就能将精力集中在各自相对独立的小型业务模块上,以实施简单建模实践,从而提高软件开发的效率。

依据业务重要性、业务稳定性、性能和成本 4 个方面总结出以下需求优先级设定方法,如表 2。并运用该方法将 8 个社保子业务分成 3 级。

表 2 需求优先级设定方法及社保业务分级

优先级	含义及相应级别的社保业务模块
1	关键性的业务,必须先实现。(档案建立、参保人员变动管理、待遇申报计算、收付计划管理、收付登记划帐、个人参保)
2	次重要性、稳定性较好的业务,即必要的系统业务,但如果需要,可以延迟实现。(年终处理)
3	次重要性、稳定性较差的业务,即最终需要而模糊的系统业务,有必要延迟实现。(权限管理)
4	性能或质量上增强;如果资源允许,实现这些需求将使软件系统更完美。

根据项目开发人员规模和项目优先级,将 8 个子业务分成 3 个开发周期。同一开发周期内,目标模块被分到各组并行开发。但各组必须经常交流,因为各模块都有相互的联系。

如果开发周期中出现新的业务需求,则按照 AM-RUP 需求建模过程,分两种情况:本周期目标模块内的新业务需求,可在 RUP 的工作流间迭代;本周期目标模块外的新业务需求,在下一周期制定迭代计划时对剩余的业务需求进行优先级的重新评审。此种方式既保证了本周期开发不受影响,又保证新业务需求得到合理的安排。

## 3.3 社保系统的需求建模——小增量迭代式简单需求建模

根据上一节制定的小增量迭代计划,开始第一个迭代周期的各小型子业务模块的简单需求建模。以下将以个人参保业务模块为例进行 AM-RUP 的需求建模阶段的实践分析。

### 3.3.1 个人参保需求探索

社保方在个人参保业务上刚刚起步,对其业务流程也并不很清楚,这就需要有一个引导用户方和开发方逐步探索需求的方法。遵循 AM-RUP 的“使用合适制品”的建模原则,本文在 AM-RUP 的需求建模过程中引入了结合场景与目标的需求分析技术(简称 SC-G 技术)。该技术是通过相互交错的目标建模和场景设置进行需求分析,它的关键结构是需求块(Requirement Chunk),定义为<目标 G,场景 SC>[2]。需求块之间的连接方式有 3 种:提取关系;组成关系(AND);可选关系(OR)。

通过 SC-G 技术的协助及双方的共同讨论,挖掘出 8 个人参保系统目标(如图 6),而且每个目标的规模已经达到了可以用一个界面来实现的程度。此处,已利用该方法的需求探索优点,完成了适度探索的任务。

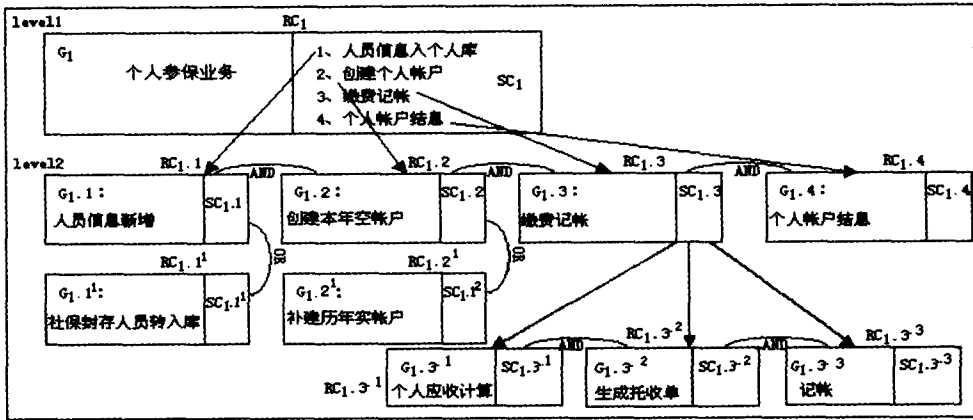


图6 个人参保需求块

3.3.2 个人参保需求详细分析

为了清晰地描述个人参保业务在以上初定的7个系统界面中如何具体实现,本文选择了AM建议的健壮图制品建模(如图7所示)。健壮图能同时描绘出边界/界面对象(用户界面元素)、实体对象(业务领域中的对象,比如参保人员)及控制/过程对象(控制前两类对象及其交互的逻辑)。比较而言,UML时序图相对要复杂,而且更适用于复杂的事物逻辑细节建模。由于此时目的是确定实现个人参保业务系统需要的各种对象及其之间的简单逻辑关系。因此,遵循AM-RUP提倡的“选择合适制品”实践,本文在此并没有选择UML时序图来进行建模。

从该健壮图中,可以很清楚了解实现个人参保业务需要

创建9个控制对象及6个实体对象、7个界面对象。虽然在深入分析设计实现时,很有可能发现需要改进,然而由于该模型较简单,可以从其它的工作流中回到需求分析 workflow 对其进行迅速讨论修改,并将修改的内容迭代到其它 workflow 当中。因此,AM-RUP提出的“创建简单的内容”、“工作流间迭代”两个实践将增强开发人员应对变化的能力。

在界面需求建模方面,可以将健壮图中的信息迭代到用户界面原型制品中,这遵循了AM的“迭代到其他制品中”实践。利用用户界面原型可以直观地与用户进行反复的界面需求讨论。用户界面原型是一种迭代技术,它不仅能进行界面需求分析,同时方便对用户的需求进行有效的验证。

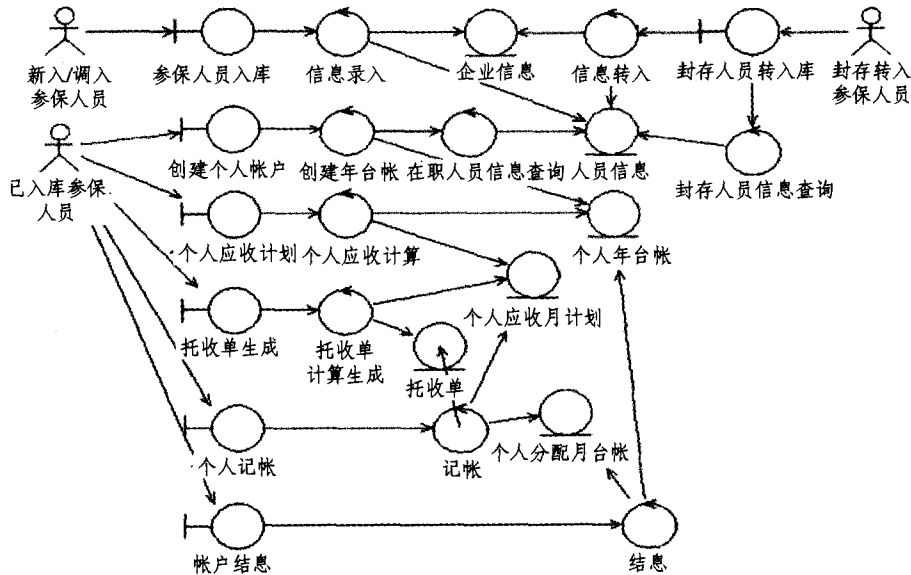


图7 个人参保业务健壮图

**结束语** 依据敏捷建模及RUP的特点,本文总结了AM和RUP结合的必要、可行性及AM-RUP的特点,并提出了一种AM-RUP需求建模过程;结合社会保险系统的部分模块的需求建模实例来具体说明如何使用敏捷化的需求建模技术及各种需求建模制品实现AM-RUP的需求建模过程,同时体现AM-RUP的需求建模过程的包容需求变化及快速反馈等特点。总之,对于RUP而言,敏捷建模是在保持其原有基本优点的前提下,加强这一重量级软件过程敏捷程度的有效途径。相信会有越来越多的统一过程软件开发实践将受益于对敏捷建模精髓的汲取。

参考文献

- 1 Ambler S W. 敏捷建模—极限编程和统一过程的有效实践[M]. 张嘉路译. 北京:机械工业出版社,2003
- 2 刘鹏辉. 敏捷方法:软件工程的新方法. 宝鸡文理学院学报(自然科学版),23(4)
- 3 董斌. 从场景中发掘目标的需求建模技术研究[J]. 安徽农业大学学报,2004,31(1)
- 4 姚恒,魏臻. 用敏捷型开发方法延长软件的生命周期[J]. 计算机时代,2004(3)
- 5 Wiegers K E. Software Requirements. Second Edition. Microsoft Press, 2003
- 6 Ambler S W. CRC Modeling, Bridging the Communication Gap Between Developers and Users. http://www.umlchina.com, 1998