

行业应用软件开发量化方法探讨^①

The Discussion of Measurement of Industry Application Software Development

古 蓉 左 春 (中国科学技术大学 230026)

摘要:本文以保险行业应用软件为例,通过对这类软件的构成、设计与实现方法的分析,讨论了软件规模的量化方法,从而帮助缺少历史数据的企业实现量化管理。

关键词:软件过程 量化管理 特征驱动

1 引言

大量实践证明,软件开发质量的提高仅有定性管理是不够的,必须要引入定量管理^[2],而量化方法则需要结合软件企业的自身情况和所开发软件的特点来设定。通过分析可以看到,行业应用软件由于其使用地域广泛,一定程度上存在着共性和个性的差异,因而从其组成上可分为横向应用、核心业务平台和个性功能脚本三大部分。在开发过程中,为了适应用户不断扩展的业务需求,将采用特征驱动开发(Feature-Driven Development, FDD)方法^[3]、自顶向下和自底向上相结合的开发模式。因此,在量化指标的选定和使用上,也分为主线和辅线两类,以适应这类软件开发过程中不同阶段的需要,保证量化过程的可操作性。

下面先简单介绍量化管理和特征驱动开发方法的相关知识,进而结合保险行业应用软件的开发实例,对行业应用软件的构成、设计与实现方法进行了分析,然后讨论了这类软件规模的量化方法并给出示例,希望对缺少历史数据的企业实现定量管理有所帮助。

2 量化管理与 FDD 方法

由定性管理向定量管理的转变是增加软件过程可视性,提高软件过程稳定性的必经之路,这个过程可以从 SEI 的 CMM 模型、CMMI 模型以及很多企业的改进成果中清楚的看到。在 CMM 模型中,量化管理随着软件企业过程能力成熟度的提高被逐步加强,CMM4 级的两个关键过程域都是围绕过程和产品的量化值来进行的。

量化管理的目的在于建立为企业自己所用的过程及产品的数据库,通过定量与定性管理的结合,达到对质量、成本、进度的控制,从而实现企业的商业目标,进入过程改进的良性循环。通过量化管理可以帮助项目管理者更好地、客观

地描绘项目的进展和工作产品的状态,有效地进行交流与沟通,准确地跟踪项目的特定目标,适时地做出必要的决策。

CMM 模型为软件企业实施过程改进提供了参考框架,但由于其更多的局限在目标和要求上,而缺少具体的改进方法,很多企业在经历了一系列的过程定义、实施、改进后,发现仍难以达到预期的目标。对项目组而言,所增加的无非是一些看似有用的文档而已,开发过程依然没有透明起来,偏差不能被及时发现,进度仍难以控制。因此,选择与开发方法相匹配的管理和量化方法是至关重要的。

软件开发方法很多,从传统的瀑布型开发到现在较为流行的增量-迭代、统一软件开发过程、FDD 方法等。FDD 方法是围绕特征进行开发的,每个特征是指小的、用户可见的功能,其特点是可以不断提交切实可行的结果^[3]。这种建立在高效沟通基础上的交付过程,使最终用户可尽早了解软件,及时修正偏差,保证了最终交付的可行性。

量化管理的基本活动在 CMM 模型中已有明确的定义,本文所讨论的内容在于以行业应用软件的构成和开发方法为基础,说明主、辅结合的量化方法。

3 行业应用软件的量化方法

行业应用软件由于使用地域广泛,不可避免会出现基础平台、个性需求上的差异,要求所开发的软件要具有较好的可扩展性和可移植性。在对这类软件进行量化前,首先要了解其特点及可能的度量方法,然后才能结合企业的开发模式和基础条件选定最终可行的量化方法。

3.1 行业应用软件开发

就行业应用软件而言,通常可以从以下几个角度来描述其构成:

(1) 从应用领域方面,可分为纵向应用和横向应用。纵

^① 该项目得到国家 863 计划项目(863-306-ZD-07-04)、国家自然科学基金重点项目(19831020)的资助

向应用面向特定的应用领域,可将其看作是业务领域相关的专用程序;横向应用可以在多个业务领域中使用,可将其视作通用的系统功能,如工具、工作流等。

(2) 从逻辑层次方面,可分为表示层、业务逻辑层和数据管理层。这种分层的处理,使系统的构架更加清晰,更好地降低各层间的耦合,利于系统的开发与维护,便于实施中应用的部署。

对于(1)中所描述的纵向应用和横向应用,都可以按这样的逻辑层次来规划。

(3) 从平台框架方面,可分为平台和脚本。这种角度的划分,更多地体现了系统的核心架构,使应用信息在共性与个性之间得以更清晰的体现。平台由纵向应用中的共性组件和横向应用组成,它们构成了软件的应用框架;而脚本则反映了纵向应用中的个性化处理,它是由纵向应用中的个性组件构成的。

图 1 是保险行业应用软件构成的示例。其中,纵向应用由承保、批改、理赔等组成,横向应用由通用工具和工作流引擎等组成。这样的划分,使得软件被应用到行业中的不同用户时,只需修改涉及脚本的部分,而核心业务平台和横向应用则相对稳定,有利于软件的推广。

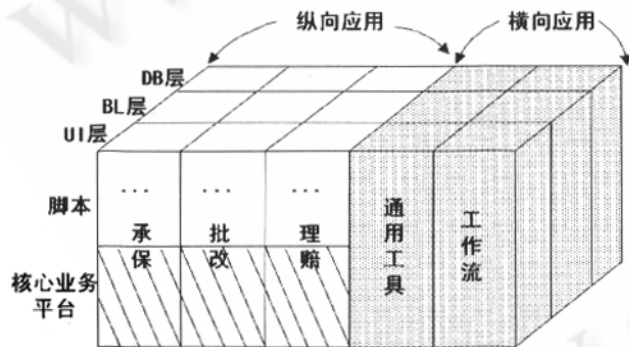


图 1 保险行业应用软件构成示例

传统的设计与实现方法可分为自顶向下和自底向上两类:

① 自顶向下:事务流程图→用例图→活动图→功能分布→操作界面→数据库表

这种设计模式是一种动态模式,它以业务需求为主线,以用例为中心,通过逐一分析设计得到所需的结果。这种以事务流程图、用例图、活动图来描述需求的方式,常被领域专家所采用。

② 自底向上:数据库表→共享数据结构→类功能区→脚本→操作界面

这种方式可看作是静态设计开发模式,通过对用户需求的归纳分析,先从数据模型的设计入手,形成数据库表的设计,然后再进行后续的共享数据结构等的设计与实现。此模式通过数据管理层到表示层的实现,完成了从单一到复合的处理。

考虑到行业应用软件这种以核心业务为平台,不断增加个性功能的开发特点,在系统框架的设计上可采用自顶向下和自底向上相结合的混合模式。在框架形成的基础上,再采用 FDD 方法,通过开发过程中的高度迭代,分批完成用户所需的个性功能要求。

在介绍了行业应用软件的构成与设计开发方法后,将讨论相关的量化方法。对于软件项目而言,基本的量化指标包括规模、成本、进度、质量等,其中,规模的度量是其他度量的基础,下面将重点介绍软件规模的量化方法。

3.2 行业应用软件的量化

软件规模是软件的重要量化指标,它的大小、估计的准确性将直接关系到软件开发的投入。因此,度量方法的选择应建立在软件开发方法的基础之上,这样,才能使估计工作有效、可行。考虑到软件规模与需求的关联性,对它的计算在设计和实现阶段完成将更符合实际情况。

通过前面对于行业应用软件的的分析,对于这类软件的规模度量可分为主线和辅线两类:

3.2.1 主线度量

主线度量是贯穿设计与开发过程始终,体现软件最终规模的度量。典型的度量元是代码行。不过,代码的编写与开发人员自身的技能、思路和习惯有很大关系,无规范的代码行度量难以客观地反映项目的规模情况,这就要求对源代码行数的统计应建立在一个标准之上。

行业应用软件的开发由于其所涵盖业务的多样性和流程处理的相似性,使得标准的建立成为可能。以保险行业软件为例,一个财产保险公司,其开办的业务可多达 20 多个险类,百余个险种,基本的业务流程都是从投保→承保→批改、理赔,其中业务的相似性在于流程和处理上,不同之处在于属性及其逻辑判断上,如图 2 所示。随着保险公司业务的扩展,险种的增加是必不可少的。同样,这类软件的开发由于其规模的庞大,往往需要由多人组成的团队共同完成,并且通常会经历较长的维护和继续开发时期。因而,在开发时如果能把握核心功能,形成统一的开发模式,使团队中每个成员的编码都似出一人,从而为人员复用、代码维护带来极大的便利。所以,为开发人员形成标准程序——“样本”程序是极为必要的。

所谓“样本”程序是指完成软件中典型应用的可行的实际标准程序集。“样本”程序与开发工具和应用领域相关,

它不是标准语句的解释性介绍,其作用在于解决行业应用软件中的相似性开发。它由一组程序构成,以编程规范为编写依据,反映了将业务模型信息化所需的各种处理,体现了软件各类组件(横向组件、核心业务平台组件和脚本)的协同工作方式,因此它区别于开发语言教科书中的范例介绍。当开发中遇到新的处理模式时,可对“样本”进行扩充,以保证其全面性。“样本”程序所对应的业务功能及其包含的每个语句都要经过严格的同行评审,使其具有应有的典型性和标准性。

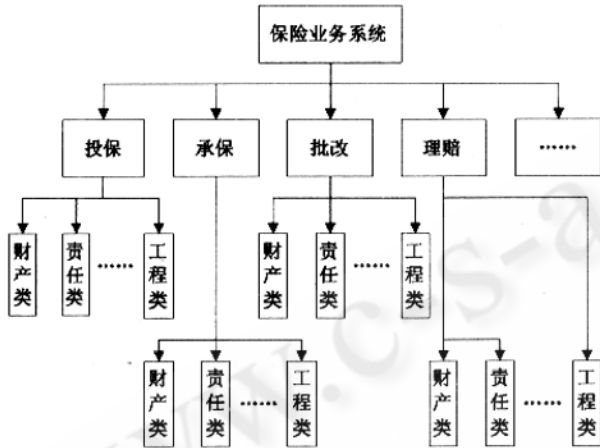


图 2 保险业务系统组成示例

保险行业应用软件的“样本”选取了财产类中财产险承保处理这个核心功能,并覆盖了界面交互、逻辑校验、后台数据交易、异常处理、事务处理、结果打印等模式。保险“样本”程序编写完成后,还利用所编制的工具反向生成了类图、活动图等文档,同时,辅助以规范检查工具,从而客户验证了“样本”程序与设计、编程规范的符合性。这样,基于前面介绍的各险种流程间的相似性,通过一个险种的承保处理程序,使开发人员不仅能了解编程的规范,还可以更深入地理解系统的框架、对象的划分、分层(表示层、业务逻辑层、数据管理层)的规划、组件的部署等,从而直观地学习和掌握开发工具,为规模化开发打下基础。当开发新险种时,往往所需修改的代码只是交互界面、个性校验和打印部分。

由于“样本”程序的以上特点,在行业应用软件开发中,它既可作为编码规范的检查标准,又可作为代码行的度量标准。同时,团队以“样本”为基础的规模化开发,以及反向工具、规范检查工具的验证,也使代码行度量更加准确。

除代码行度量外,在主线索度量中还可以度量功能数和类数。代码行的度量展现了软件最终的实现结果,而功能数和类数也从设计实现的角度反映出了软件的规模。以功能数为基础的度量,可以从用户可见的角度来反映软件,这与

FDD 的开发方法相适应;而类的度量则反映了面向对象的设计结果。

因此,以代码行、功能数、类数为主线的度量,可以从不同的角度和粒度来度量软件的规模,较好地刻画出软件设计与实现之间的规模关系。

3.2.2 辅线度量

辅线度量是在开发过程中,以阶段成果为基础进行的度量,用以辅助了解开发过程中的软件规模。对于以数据库为基础的行业应用软件,面向数据结构和用户交互界面的度量是非常必要的,为此可采用结构数、界面数作为辅线度量元。

这里所指的结构,是指与数据库表相关联的字段集,可分为基础结构和复合结构两类。其中,基础结构与数据表结构对应,复合结构则根据特殊处理要求选定。这个指标反映了软件对数据表读写的集合,它的变更仅发生在数据结构发生调整的情况下,因而,在开发过程中它具有一定的稳定性,对它的度量可从侧面展现软件的规模。

行业应用软件中用户交互界面的设计与实现是非常庞大的一项工作,对它的度量可很好地帮助项目组规划软件的开发,因此它也是一个有效的辅助度量元。

考虑到行业软件框架与个性功能脚本开发的差异,在规模度量上可选取不同的指标:

- 软件框架

软件框架由横向应用和核心业务平台组成,这部分内容与用户的直接交互较少,可选择结构数来辅助标识规模。

- 脚本

脚本的开发是面向用户的可见功能进行的,对它的度量除了软件框架可用的度量指标外,界面数也是非常重要的一项。

通过对结构数和界面数的度量,从侧面反映出开发任务及其复杂性,为后续的工作量度量提供了更多的参考信息。

3.3 实例

这里以保险行业应用软件新增险种为例说明规模的度量结果,见表 1。根据前面的讨论,已构建的横向应用、核心业务平台以源代码行、类数为主线度量,以结构数为辅线度量;脚本则从源代码行数、类数、功能数、结构数、界面数分别进行了主线和辅线的度量。表中已有险种脚本的结构数是在核心业务平台上所进行的扩充部分。对于后续开发的新险种的规模度量,只体现在在已有平台和脚本的基础上所进行的个性化开发的部分。表 1 中所列的新险种“火灾公众责任险”的开发复用了已有的结构,代码的修改主要集中在界面校验、提示和打印上,因此结构数的变更为 0,相关的界面变更为 6 个,类的变更为 18 个,代码行的变更较少。

表 1 规模度量示例

统计对象	主线度量			辅线度量	
	源代码 (行)	类 (个)	功能 (个)	结构 (个)	界面 (个)
横向应用	150000	450		135	
核心业务平台	320000	310		201	
已有险种脚本	280000	290	560	93	1565
火灾公众责任险	900	18	5		6

从实际工作来看,软件规模的大小及软件的复杂度直接关系到工作量的大小,通过对软件规模的分类度量,可以从多角度量化软件,为进度的安排提供了基础数据。

4 结束语

对于软件的量化,不同的企业可以采用不同的方法和量化指标,但关键在于过程控制要与方法学相结合,并能准确描绘软件过程和产品的真实状态,满足不同管理层次人员的需求,这样才能增加软件过程的透明性,真正达到量化管理

的目的。本文所介绍的行业软件开发的量化方法,还需在度量的自动化、变更管理和问题管理上加以改进,必要时引入新的量化指标,从而更全面的度量软件。

参考文献

- 1 Mark C Paulk, Bill Cuntis, et al. Capability Maturity Model for Software, Version 1.1 [R]. Technical Report CMU/SEI-93-TR-024, ESC-TR-93-177, February 1993. 9-14, 30-43.
- 2 吴超英、廖彬山译, [美] John McGarry 等著, 实用软件度量[M], 机械工业出版社, 2003.
- 3 熊焕宇等译, [英] Stephen R. Palmer, John M. Felsing 著, 特征驱动开发方法原理与实践[M], 机械工业出版社, 2003.
- 4 钱岭、苏薇、盛轶阳译, [美] David Garmus, David Heron 著, 功能点分析 [M], 清华大学出版社, 2003.
- 5 胡春哲等译, [印] Pankaj Jalote 著, CMM 实践应用——Infosys 公司的软件项目执行过程 [M], 电子工业出版社, 2002.