

软件开发过程的选择模型

(Model of Selection of Software Development Process)

黎 闯 (广州暨南大学计算机系 510632)

摘要:软件开发过程的选择是各软件开发组织在开发软件时首先遇到的问题,不同的开发过程适用于不同的软件和软件开发组织。分析影响开发过程选择的主要因素,总结出一个开发方法的选择模型。

关键词:软件开发过程 模型选择 过程模型

为了解决“软件危机”,人们提出了软件工程的观念,对软件整个生命周期进行管理。在软件生命周期中,软件开发过程占主要部分。经过软件工程的实践,人们总结出了一系列开发模型。通过这些模型的运用和研究,又不断改进和提出新的模型,如现在研究较热门的 CMM, RUP, Agile Development 等。但各种模型都有其优缺点,都只是适应某类具体情况,没有统一的都可适用的模型。因此,对模型的选择就显得很重要。选择正确的适用具体开发工程项目情况的软件开发过程,是保证软件开发成功的第一步。

Thayer 等在他们的一篇文章中^[1]列出了 20 个软件管理者所提出的主要问题,其中超过一半涉及到软件开发技术的选择。一个具体的软件过程包括开发过程,及所用方法和工具。R. Basili 等在^[2]中提出了一个量化方法,在相应的开发环境下选择开发方法和工具。但开发过程的选择更需要关心。一个适合的开发过程应该是基于待开发系统的性质及其与开发环境之间的关系^[3],通过对这两方面的分析,可以找出一条指导性的途径,选择开发过程。

1 典型软件开发过程模型

(1) code and fix。最简单、普遍、使用最广泛的软件开发方法。即通常所用的方法,不经过设计,直接进行编程。错误在所难免,一般到后期才检查纠正错误。

(2) 瀑布模型。由 W. W. Rocy 经过大量的工程实践于 1970 年提出。其开发过程固定地顺序经过需求分析,设计,实现,测试和维护几个步骤和阶段。每一阶段以前一阶段的开发成果为开发依据。

(3) V—模型。瀑布模型的一种变体。给每一层开发活动增加了测试和验证,每进一层的开发,都使系统模型更加详细,测试也从低层到高层逐渐进行。

(4) 螺旋模型。Boehm 针对瀑布模型的缺点提出的改进模型。是一种循环式开发模型。每一循环包括四个相同阶段,即确定目标,选择评估,进行相应开发下一周期计划。不断循环,直至得到满意产品。

(5) 原型模型。根据已知的系统需求,开发一个简化版的系统,实现系统的主要功能。用户据此反馈回所需增加的需求及进一步可明确的需求,开发者继续修改和增加原型系统,直至明确用户全部需求。

(6) RAD。原型系统的改进模型。每一个开发阶段分成固定的时间片,每个时间片完成一定的内容,这样能在规定的时间内完成预定的内容。即对开发时间进行严格限制。

(7) Synch - and - Stabilize。微软所使用的方法。不同的小型开发小组进行同步开发,快速开发、快速检测,以期在较短时间内拿出产品,占领市场。然后再改进产品,生产更新和升级版本。

(8) 增量和迭代开发。典型的如:Rational 的统一软件开发过程(RUP)。将大的软件开发项目分成一些小的项目,每个小项目作为一次迭代,对目前最重要的开发风险进行评估和处理,得到进一步的开发成果,使系统得以不断的扩展,逐步增加的开发过程最终实现整个系统。

(9) 探索模型(Exploratory Model)。在系统需求很难明确,或者根本不知道的情况下,从已知的较少信息出发构造系统,对系统进行测试,根据结果进行改

进,重复上述过程,直到能得出满意的结果。

(10) 基于实体的模型。有如:基于问题的开发(Issue-based development)和基于工作簿的开发(Workbook-based development)。把“问题”或“工作簿”作为开发的依据,围绕“问题”可以实现并行开发,而“工作簿”则是作为项目成员间的通信工具。

(11) 能力成熟度模型。SEI(软件工程组织)将软件开发组织的“开发能力成熟度”分成五个层次,用于指导软件开发组织提高软件开发能力,同时找到适合本组织的软件开发过程。更多意义上,能力成熟度模型不是普通意义上的开发模型,但通过它的指引,可以使软件开发组织采用更利于成功的开发过程。

(12) Agile Development。即目前研究较热的“敏捷软件开发”。基本思想是充分利用用户的参与,在较短的时间内迅速开发出所需的某项功能,保证所实现的部分经过充分测试无错。通过用户及开发小组成员的充分交流,确保用户需求得以明确并正确理解。在每次都开发充分的情况下,经过多次循环,最终开发出所需系统。

这些是目前软件开发中主要使用的模型,也就是进行分析的基础。

2 项目性质和开发环境对软件开发过程选择的影响

主要分析:待开发软件的规模,类型,质量要求,交付工期,需求明确度,以及投资,可重用,开发者掌握资源等。因为这些反映了项目软件的基本特性,以及开发环境影响开发过程的基本方面,是影响开发过程选择的主要因素:

2.1 软件规模

大规模系统就不能用第(1)种开发过程,因为这样将无法进行管理。对小规模的系统,由于复杂性有限,从实践、成本等方面考虑,采用较复杂的过程可能没有必要,一般都会用第(1)种方法。现实中绝大部分个人开发的小软件都是采用这种方法。实际上,由于系统规模小,对其测试也较方便,也容易对质量进行控制。同时,比较时兴的开发方法如 XP(eXtreme Programming,属 Agile Development 主要部分)也在目前运用较多,因为 XP 也是运用于中小系统,而且很有效。

2.2 软件类型

Pressman 提出了八个不同的软件领域,不同领域的软件有其不同的特点,运用不同的开发过程。

(1) 系统软件。系统软件的开发,需要在工具的支持下进行。而且如操作系统等,需要解决多进程的死锁等问题,要使用形式化方法。遵从瀑布模型仍然是一个好的选择,也可以同其他新型开发方法结合使用。

(2) 实时软件。由于有严格的时间特性,实时软件一般对质量要求较高,特别是关系到人身、财产安全的系统,如医疗,军事等方面的应用。所以实时软件开发一般要用形式化方法,保证软件的质量。这样开发的难度会增加,时间也会更长,相应费用也会增加。应该选用对这些不太敏感的过程。

(3) 商业运用软件。商业信息处理是最大的单个软件应用领域。各业务领域各自特性很强,领域内的组件可重用性很强,重用组件可达到很高的效率。又由于企业业务经常都会变,用面向对象技术和可重用技术,可以很好地把修改局限在一个小范围内。因此,利用可重用技术是首选考虑。同时,由于原型开发中从原型到最终产品的转化耗费巨大,所以一般在此领域也不使用原型开发模型。需要与领域专家的交流限制了形式化方法的使用。除非质量要求很高,一般不使用形式化方法。目前 RUP 和 XP 技术就比较适合于这方面开发。

(4) 现代工程和科学计算。大量的是数值处理。由于需求较明确,可以采用传统的如瀑布模型,V—模型等开发过程。

(5) 嵌入式。嵌入式软件是指嵌入到电子产品中控制其运行的软件。嵌入式软件的开发一般需要专门的开发工具。嵌入式软件开发环境是一个集成的软件开发工具,用来管理整个嵌入式软件的开发过程。一般可用传统的瀑布模型,V—模型等。

(6) 个人计算机。微软的“Synch-and-Stabilize”模式无疑应该是作为个人计算机软件商业开发的良好模式。但除了操作系统外,其上还有成千上万的应用。随着个人计算机的普及,其应用还在扩大。对一些简单、小的应用,使用第(1)种方法也是可行的。

(7) Web 应用。随着网络的发展,该方面应用迅速增长。由于很多涉及到跨平台操作,其开发与传统软件有不同之处。现在对 UML 应用和 JAVA 等跨平台语言的研究愈来愈深入,且有众多的工具支持,基于

UML 的 RUP 给该领域软件开发提供了很好的方向。

(8) 人工智能。软件开发初期很难定义系统需求,因为人工智能方面的系统都是基于猜想、估计和假设。一般都是先假设系统的功能和流程,据此快速开发出系统,再根据对系统的评估进行改进,经过多次循环,最后得到满意的产品。这个领域的系统用别的开发过程都不容易做到。

2.3 软件质量要求

安全性、稳定性要求高的不能用第(1)种方法,因为这种方法完全靠开发者个人的能力来保证软件的质量,软件的好坏全靠个人的发挥,无法保证能生产出质量恒定的产品。可使用统一软件开发过程或能力成熟度模型以及敏捷软件开发技术(Agile Development)。其中的XP技术就是所谓的“测试驱动”,以产生高质量程序为目的。这几种模型在保证软件的质量方面,经实践证明有较好的效果。还有“净室技术”,该技术用形式化方法进行系统说明,并对程序进行正确性证明,产生所谓“零缺陷程序”,能得到高质量的软件。

2.4 交付工期

用户对交付期要求严格的项目,可以采用快速原型法即RAD法,它将每个开发循环分成固定的时间片,每个时间片完成一定量的需求,实现一个原型,经过若干个循环,最后完成完整的产品。该模型按时间来安排项目开发,有利于对开发时间的掌握。同时,也可以考虑用螺旋模型。因为螺旋模型每次循环都包括风险评估阶段,会对下一阶段的实现进行时间上的评估。如果估计所需时间超出规定期限,项目将中止。只不过这样将得不到所需产品,而且对时间的管理也是估计性的,不是从时间角度出发。

2.5 用户需求明确度

主要影响选用开发方式的要素。

由于瀑布模型的缺点之一便是不能适应用户需求变化,必须是用户需求很明确,并且开发人员能明确理解的前提下,可考虑用瀑布模型。对在开发初期需求不是很明确,或者需求一直在变的项目,可采用原型和增量开发的方法,经过不断的循环,逐步明确用户的需求,直至开发出用户满意的产品。对于需求不确定,或者需求还根本不知道的情形,可以使用快速原型法,其中利用可重用技术,用已有的组件、框架、代码,快速实

现原型,并根据用户反馈,明确或调整系统需求,最终开发出用户所要求的产品。XP技术也是在需求变化情况下,很好的方法。

2.6 投资

用探索模型开发的系统就很难进行成本效益核算,故用户对投资有严格要求的项目不能用此方法。在螺旋模型中,每次风险评估有对进一步的实现做投资方面的评估。如果下一步的投资评估超过了用户限定,或者终止项目开发,或者经用户同意追加投资。

2.7 可重用

重用可以节约时间和花费。开发软件系统最难以实现的目标就是达到高层次的重用。而面向对象技术最主要的好处就在于代码和设计的重用,这一点现在几乎大家已经达成了共识。其他技术的可重用讨论的也很多,但到实际运用还有距离。目前阶段可重用技术都是和面向对象联系在一起。即开发者要根据已有的和可获得的重用构件,以及用户其他要求,开发系统的特性等决定是否采用可重用技术,以及与可重用技术相关的开发过程,提高效率。

2.8 开发者掌握资源

对开发过程的选取影响是综合性。包括人员、技术、工具以及软件资源等几个方面。人员方面考虑的主要因素包括:知识、技术、经验、能力。开发人员所具有的开发项目相关领域的知识,所需技术掌握程度,开发经验,以及综合方面的能力,会对开发过程选择产生影响。另外,开发组织采用的技术标准,拥有的开发工具,也是决定所用模型的重要因素。开发组织已有的可重用资源,包括组件、框架、设计模式等,也是考虑的重要内容。

3 开发过程选择模型

根据以上分析,开发过程的选择受很多因素影响,选择中开发者考虑的侧重点不同,得出的结果也就不同。各因素间是并行的,普遍情况下没有先后、主次之分。故在选择同时考虑,一种选择的参考模型为:

(1) 根据待开发项目情况从下表列出可用模型:(表1)

(2) 对本开发组织所拥有资源进行评估,从上一步所列出的可用模型中选择最适合当前开发项目的开发过程;

表 1 项目特性与对应开发模型

项目	推荐模型(用序号表示)	理由
软件规模	小	1,2,3,12
	大	除 1,12
软件类型	系统软件	2,3,4,8,11
	实时软件	8,11,12
	商业运用软件	8,11,12
	现代工程和科学计算	2,3,11
	嵌入式	2,3
	个人计算机	1,4,5,7,10,11,12
	Web 应用	8,11
	人工智能	6,9
对软件质量要求	低	不影响
	高	8,11,12
工期要求	不严格	不影响
	严格	4,6
需求明确度	不明确,在变	5,6,8,12
	明确,不变	2,3,4
对投资限定	不严格	不影响
	严格	4
可重用技术	不采用	不影响
	采用	6,8,11,12

4 结论

软件项目开发中,开发过程的选择既重要又比较困难。有很多因素影响到开发过程的选择,主要包括:软件规模,类型,交付日期,用户需求明确度,投资,可重用,开发者掌握的资源。它们分别从不同方面对开发选择产生影响,需要综合考虑。通过对这些主要因素的分析,可以找出一条选择的途径,作为开发者进行选择时的一个指导。需要指出的是,所选出的适合模型只是一个参考模型,还须与具体开发环境结合,对模型进行裁剪,才能得到具体工程中所使用的开发过程。

参考文献

- 1 R. H. Thayer, A. Pyster, and R. C. Wood. The Challenge of Software Engineering Project Management [J]. IEEE Computer Magazine, vol. 13, no. 8, August 1980: pp 51 – 59.
- 2 V. R. Basili and H. D. Rombach. Tailoring the Software Process to Project Goals and Environments [J]. Proceedings of the Ninth International Conference on Software Engineering, Monterey, CA (March 30 – April 30 – April 2, 1987): pp. 345 – 357.
- 3 Frank Kand. A Contingency Based Approach to Requirements Elicitation and Systems Development [J]. London School of Economics, J. Systems Software, 1998, 40: pp. 3 – 6.