

软件过程

本部分将介绍软件过程，它提供了软件工程实践的框架。在本部分的各章中将涉及以下问题：

- 什么是软件过程？
- 软件过程中，有哪些通用的框架活动？
- 如何建立过程模型？什么是过程模式？
- 什么是惯用过程模型？有哪些优缺点？
- 为什么现代软件工程关注敏捷问题？
- 什么是敏捷软件开发？它与传统的过程模型有什么区别？

在解决了上述问题之后，就可以对软件工程实践的应用背景有更清楚的认识。

软件过程结构

要点浏览

概念: 在开发产品或构建系统时, 遵循一系列可预测的步骤(即路线图)是非常重要的, 它有助于及时交付高质量的产品。软件开发中所遵循的路线图就称为“软件过程”。

人员: 软件工程师及其管理人员根据需要调整开发过程, 并遵循该过程。除此之外, 软件的需求方也需要参与过程的定义、建立和测试。

重要性: 软件过程提高了软件工程活动的稳定性、可控性和有组织性, 如果不进行控制, 软件活动将变得混乱。但是, 现代软件工程方法必须是“灵活”的, 也就是要求软件工程活动、控制以及工作

产品适合于项目团队和将要开发的产品。

步骤: 具体来讲, 采用的过程依赖于构造软件的特点。飞机航空系统的软件与网站的建设可能需要采用两种截然不同的软件过程。

工作产品: 从软件工程师的角度来看, 工作产品体现为在执行过程所定义的任务和活动的过程中, 所产生的程序、文档和数据。

质量保证措施: 有大量的软件过程评估机制, 开发机构可以评估其软件过程的“成熟度”。然而, 表征软件过程有效性的最好指标还是所构建产品的质量、及时性和寿命。

Howard Baetjer Jr.[Bae98] 曾著书从经济学家的角度分析软件和软件工程, 该书引人入胜, 对软件过程评述如下:

软件同其他资产一样, 是知识的具体体现, 而知识最初都是以分散的、不明确的、隐蔽的且不完整的形式广泛存在的, 因此, 软件开发是一个社会学习的过程。软件过程是一个对话的过程, 在对话中, 获取需要转化为软件的知识, 并在软件中实现这些知识。软件过程提供了用户与设计人员之间、用户与不断演化的工具之间以及设计人员与不断演化的工具(技术)之间的互动。软件开发是一个迭代的过程, 在其中演化的工具本身就作为沟通的媒介, 任何新一轮对话都可以从参与的人员中获得更有用的知识。

构建计算机软件确实是一个迭代的社会学习的过程, 其输出——即 Baetjer 所称的“软件资产”——是知识的载体, 这些知识在过程执行中进行收集、提炼和组织。

但从技术的角度如何确切地定义软件过程呢? 本书将软件过程定义为一个为创建高质量软件所需要完成的活动、动作和任务的框架。过程与软件工程同义吗? 答案是“是, 也不是”。软件过程定义了软件工程化中采用的方法, 但软件工程还包含该过程中应用的技术——技术方法和自动化工具。

更重要的是, 软件工程是由有创造力、有知识的人完成的, 他们根据产品构建的需要和

关键概念

- 通用过程模型
- 过程评估
- 过程流
- 过程改进
- 过程模式
- 任务集

市场需求来选取成熟的软件过程。

3.1 通用过程模型

在第2章中，过程定义为在工作产品构建过程中所需完成的工作活动、动作和任务的集合。这些活动、动作、任务中的每一个都隶属于某一框架或者模型，框架或模型定义了它们与过程之间或者相互之间的关系。

软件过程示意图如图3-1所示。由图可以看出，每个框架活动由一系列软件工程动作构成；每个软件工程动作由任务集来定义，这个任务集明确了将要完成的工作任务、将要产生的工作产品、所需要的质量保证点，以及用于表明过程状态的里程碑。

关键点 在软件过程中，技术工作的层次包括活动，活动由动作构成，动作由任务构成。

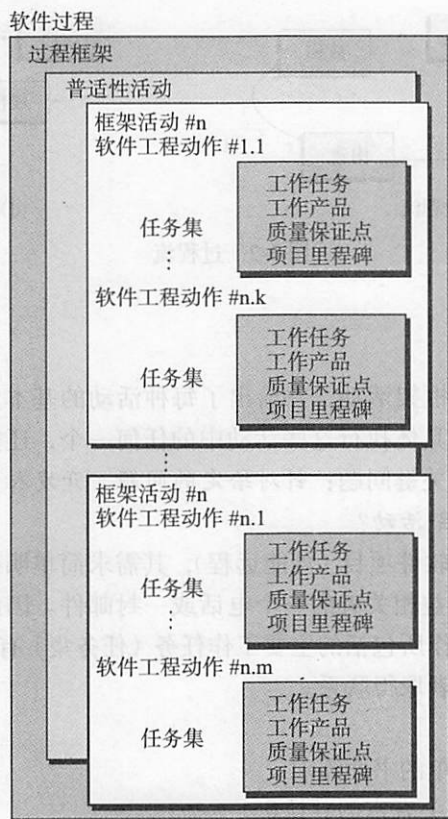


图 3-1 软件过程框架

正如在第2章中讨论的，软件工程的通用过程框架定义了五种框架活动——沟通、策划、建模、构建以及部署。此外，一系列普适性活动——项目跟踪控制、风险管理、质量保证、配置管理、技术评审以及其他活动——贯穿软件过程始终。

你也许注意到了，软件过程的一个很重要的方面还没有讨论，即过程流（process flow）。过程流描述了在执行顺序和执行时间上如何组织框架中的活动、动作和任务，如图3-2所示。

提问 什么是过程流？

线性过程流（linear process flow）从沟通到部署顺序执行五个框架活动（参见图3-2a）。迭代过程流（iterative process flow）在执行下一个活动前重复执行之前的一个或多个活动

(参见图 3-2b)。演化过程流 (evolutionary process flow) 采用循环的方式执行各个活动, 每次循环都能产生更为完善的软件版本 (参见图 3-2c)。并行过程流 (parallel process flow) (参见图 3-2d) 将一个或多个活动与其他活动并行执行 (例如, 软件一个方面的建模可以同软件另一个方面的构建活动并行执行)。

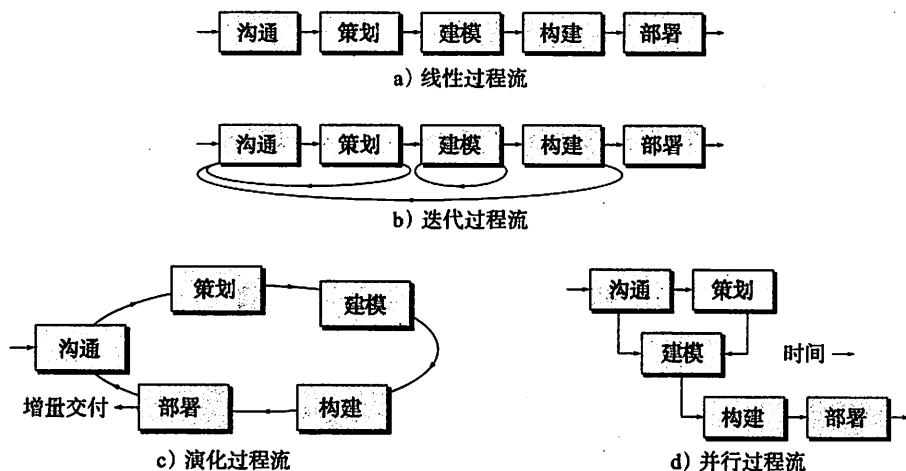


图 3-2 过程流

3.2 定义框架活动

尽管第 2 章描述了 5 种框架活动, 并给出了每种活动的基本定义, 但是软件团队要在软件过程中具体执行这些活动中的任何一个, 还需要更多信息。因此, 我们面临一个关键问题: 针对给定的问题、开发人员和利益相关者, 哪些动作适合于框架活动?

对于由个人负责的小型软件项目 (可能远程), 其需求简单明确, 沟通活动也许仅仅是与合适的利益相关者的一个电话或一封邮件。因此, 主要的动作是电话交流, 这个动作所包括的主要工作任务 (任务集) 有:

1. 通过电话与利益相关者取得联系。
2. 讨论需求并做记录。
3. 将笔记整理成一份简单的书面需求。
4. 通过 E-mail 请利益相关者审阅并认可。

如果项目有多个利益相关者, 则要复杂得多, 每个参与人员都有着不同的需求 (有时这些需求甚至是相互冲突的), 沟通活动可能会包含 6 个不同的动作 (具体参见第 8 章): 起始、需求获取、需求细化、协商、规格说明和确认。每个软件工程动作都可能有很多工作任务和一些不同的工作产品。

3.3 明确任务集

我们再来看图 3-1, 每一个软件工程动作 (如需求获取以及与沟通活动相关的动作) 都由若干个任务集 (task set) 构成, 而每一个任务集都由软件工作任务、相关工作产品、质量保证点和项目里程碑组成。需要选择最能满足项目需要并适合开发团队特点的任

引述 如果过程正确, 就会得到应有的结果。

Takashi Osada

提问 框架活动如何随着项目性质的变化而变化?

关键点 不同的项目需要不同的任务集。软件团队根据问题和项目的特点选择任务集。

务集。这就意味着软件工程动作可以根据软件项目的特定需要和项目团队的特点做适当的调整。

信息栏 任务集

任务集定义了为达到一个软件工程动作的目标所需要完成的工作。例如，需求获取（通常称为“需求收集”）就是发生在沟通活动中的一个重要的软件工程动作。需求获取的目的是理解利益相关者对将构建的软件的需求。

对于一个小型、相对简单的项目而言，需求获取的任务集可能包括：

1. 制定项目的利益相关者列表。
2. 邀请所有的利益相关者参加一个非正式会议。
3. 征询每个人对于软件特性和功能的需求。
4. 讨论需求，并确定最终的需求列表。
5. 划定需求优先级。
6. 标出不确定域。

对于大型、复杂的软件工程项目而言，可能需要如下不同的任务集：

1. 制定项目的利益相关者列表。
2. 和利益相关者的每个成员分别单独讨论，获取所有的要求。

3. 基于利益相关者的输入，建立初步的功能和特性列表。
4. 安排一系列促进需求获取的会议。
5. 组织会议。
6. 在每次会议上建立非正式的用户场景。
7. 根据利益相关者的反馈，进一步细化用户场景。
8. 建立一个修正的利益相关者需求列表。
9. 使用质量功能部署技术，划分需求优先级。
10. 将需求打包以便于软件可以实施增量交付。
11. 标注系统的约束和限制。
12. 讨论系统验证方法。

上面两种任务集都可以完成需求获取，但是无论从深度还是形式化的程度上来说，二者都有很大区别。软件团队采取适当的任务集以达到每个动作的目的，并且保持软件质量和开发的敏捷性。

34

3.4 过程模式

每个软件团队在软件过程里都会遇到很多问题。针对这些问题，如果软件团队能够得到已有的经过验证的解决方案，将有助于他们快速地分析和解决问题。过程模式（process pattern）^①描述了软件工程工作中遇到的过程相关的问题，明确了问题环境并给出了针对该问题的一种或几种可证明的解决方案。通俗地讲，过程模式提供了一个模板 [Amb98]——一种在软件过程的背景下统一描述问题解决的方法。通过模式组合，软件团队可以解决问题并定义最符合项目需求的开发过程。

我们可以在不同抽象层次上定义模式^②。在某些情况下，模式可以描述一个与完整过程模型（例如原型开发）相关的问题（及其解决方案）；在其他的情况下，模式可以描述一个与

提问 什么是过程模式？

① 关于模式的详细讨论参见第11章。

② 模式的概念广泛应用于软件工程的活动中。第11、13、15、16和20章将分别讨论分析模式、设计模式和测试模式。本书第四部分将讨论项目管理活动中的模式和反模式。

框架活动（如策划）或者框架活动中的一动作（如项目估算）相关的问题（及其解决方案）。

Ambler[Amb98] 提出了下面的过程模式的描述模板：

模式名称。模式名称应能清楚地表述该模式在软件过程中的含义（例如技术评审）。

驱动力。模式的使用环境及主要问题，这些问题会显现在软件过程中并可能影响解决方案。

类型。定义模式类型。Ambler[Amb98] 提出了三种类型：

1. **步骤模式（stage pattern）**——定义了与过程的框架活动相关的问题。

由于框架活动包括很多动作和工作任务，因此步骤模式包括与步骤（框架活动）有关的许多任务模式（见以下描述）。例如，建立沟通可能作为一个步骤模式，该步骤模式可能包括需求收集等任务模式。

2. **任务模式（task pattern）**——定义了与软件工动作或是工作任务相关、关系软件工程实践成败的问题（例如，需求收集是一个任务模式）。

3. **阶段模式（phase pattern）**——定义在过程中发生的框架活动序列，即使这些活动流本质上是迭代的。例如，螺旋模型和原型开发^①就可能是两种阶段模式。

启动条件。它描述的是模式应用的前提条件。在应用模式之前需要明确：（1）在此之前，整个开发组织或是开发团队内已经有哪些活动？（2）过程的进入状态是什么？（3）已经有哪些软件工程信息或是项目信息？

例如，策划模式（阶段模式）需要的前提条件有：（1）客户和软件工程师已经建立了合作的交流机制；（2）已经成功完成一些客户沟通模式中特定的任务模式；（3）项目范围、基本业务需求和项目限制条件已经确定。

问题。描述模式将要解决的具体问题。

解决方案。描述如何成功实现模式。这部分主要讨论随着模式的启动，过程的初始状态（模式应用之前就已经存在）是如何发生改变的。解决方案也描述了随着模式的成功执行，模式启动之前所获得的软件工程信息和项目信息是如何变换的。

结果。描述模式成功执行之后的结果。模式完成时需要明确：（1）必须完成哪些开发组织或是开发团队相关的活动？（2）过程的结束状态是什么？（3）产生了哪些软件工程信息或是项目信息？

相关模式。以层次化或其他图的方式列举与该模式相关的其他过程模式。例如步骤模式沟通包括了一组任务模式：项目团队组织、合作指导原则定义、范围分解、需求收集、约束描述以及场景模式的创建等。

已知应用和实例。说明该模式可应用的具体实例。例如，沟通在每一个软件项目的开始都是必需的，建议在整个软件项目过程中采用，并规定在部署活动中必须进行。

过程模式提供了一种有效的机制，用以解决任何与软件过程相关的问题。模式使得软件工程组织能够从高层抽象开始（阶段模式）建立层次化的过程描述。高层抽象描述又进一步细化为一系列步骤模式以描述框架活动，然后每一个步骤模式又进一步逐层细化为更详细的任务模式。过程模

引述 模式的重复与软件模块的重复完全不同。事实上，不同的模块是独特的，而模式却是相同的。

Christopher
Alexander

关键点 模式模板提供了描述模式的一般性方法。

引述 我们认为软件开发人员遗漏了一个事实：多数机构并不清楚他们在做什么，他们以为清楚了，但实际上不清楚。

Tom DeMarco

网络资源 可在下面的网站查看过程模式的全面资源：
www.amblysoft.com/processPatternsPage.html。

① 第4章将讨论这些阶段模式。

式一旦建立起来,就可以进行复用以定义各种过程变体,即软件开发团队可以将模式作为过程模型的构建模块,定制特定的过程模型。

信息栏 过程模式实例

当利益相关者对工作成果有大致的想法,但对具体的软件需求还不确定时,下述简化的过程模式描述了可采用的方法。

模式名称。需求不清。

目的。该模式描述了一种构建模型(或是原型系统)的方法,使得利益相关者可以反复评估,以便识别和确定软件需求。

类型。阶段模式。

启动条件。在模式启动之前必须满足以下四个条件:(1)确定利益相关者;(2)已经建立起利益相关者和软件开发团队之间的沟通方式;(3)利益相关者确定了需要解决的主要问题;(4)对项目范围、基本业务需求和项目约束条件有了初步了解。

问题。需求模糊或者不存在,但都清楚地认识到项目存在问题,且该问题需要

通过软件解决。利益相关者不确定他们想要什么,即他们无法详细描述软件需求。

解决方案。描述了原型开发过程,详见4.1.3节。

结果。开发了软件原型,识别了基本的需求(例如交互模式、计算特性、处理功能等),并获得了利益相关者的认可。随后,可能有两种结果:(1)原型系统可以通过一系列的增量开发,演化成为软件产品;(2)原型系统被抛弃,采用其他过程模式建立了产品软件。

相关模式。以下模式与该模式相关:客户沟通,迭代设计,迭代开发,客户评价,需求抽取。

已知应用和实例。当需求不确定时,推荐原型开发方法。

3.5 过程评估与改进

软件过程并不能保证软件按期交付,也不能保证软件满足客户要求,或是软件具备了长期质量保证的技术特点(第19章)。软件过程模型必须与切实的软件工程实践相结合(本书第二部分)。另外,对过程本身也要进行评估,以确保满足了成功软件工程所必需的基本过程标准要求^①。

在过去的几十年中,人们提出了很多种不同的软件过程评估和改进方法。

用于过程改进的CMMI[®]标准评估方法(Standard CMMI Assessment Method for Process Improvement, SCAMPI)——提供了五步的过程评估模型:启动(initiating)、诊断(diagnosing)、建立(establishing)、执行(acting)和学习(learning)。SCAMPI方法采用SEI的CMMI作为评估的依据[SEI00]。

用于组织内部过程改进的CMM评估(CMM-Based Appraisal for Internal Process Improvement, CBA IPI)——采用SEI的CMM作为评估的依据[Dun01],提供了一种诊断方法,用以分析软件开发机构的相对成熟度。

SPICE (ISO/IEC 15504)——该标准定义了软件过程评估的一系列要求。该标准的目的是帮助软件开发组织建立客观的评价体系,以评估定义

关键点 以改进为目标,评估力求理解软件过程的当前状态。

37

引述 软件组织还存在很大的能力缺陷,难以将其从整个项目中所获得的经验转化成资产。

NASA

① SEI CMMI[CMM07]丰富详实地介绍了软件过程的基本特征,以及过程成功的标准。

② 有关CMMI的详细介绍参见37.3节。

的软件过程的有效性 [ISO08]。

软件 ISO 9001:2000——这是一个通用标准，任何开发组织如果希望提高所提供的产品、系统或服务的整体质量，都可以采用这个标准。因此，该标准可直接应用于软件组织和公司 [Ant06]。

有关软件评估和过程改进方法的详细讨论参见第 37 章。

3.6 小结

一个软件工程通用过程模型包含了一系列的框架和普适性活动、动作以及工作任务。每一种不同的过程模型都可以用不同的过程流来描述，工作流描述了框架活动、动作和任务是如何按顺序组织的。过程模式用来解决软件过程中遇到的共性问题。

习题与思考题

- 3.1 在本章的介绍中，Baetjer 说过：“软件过程提供了用户与设计人员之间、用户与开发工具之间以及设计人员与开发工具之间的互动。”对以下四个方面各设计五个问题：（1）设计人员应该问用户的；（2）用户应该问设计人员的；（3）用户对将要构建的软件的自问；（4）设计人员对于软件产品和建造该产品采取的软件过程的自问。
- 3.2 讨论 3.1 节所描述的不同过程流之间的区别。你是否能够确定适用于所描述的每种通用流的问题类型？
- 3.3 为沟通活动设计一系列动作，选定一个动作为其设计一个任务集。
- 3.4 在沟通过程中，遇到两位对软件如何做有着不同想法的利益相关者是很常见的问题。也就是说，你得到了相互冲突的需求。设计一种过程模式（可以是步骤模式），利用 3.4 节中针对此类问题的模板，给出一种行之有效的解决方法。

扩展阅读与信息资源

大多数软件工程课本都会详细介绍过程模型。Sommerville（《Software Engineering》，9th ed., Addison-Wesley, 2010）、Schach（《Object-Oriented and Classical Software Engineering》，8th ed., McGraw-Hill, 2010）以及 Pfleeger 和 Atlee（《Software Engineering: Theory and Practice》，4th ed., Prentice Hall, 2009）的书中介绍了这些传统的模型，并讨论了它们的优点和缺点。Munch 和他的同事（《Software Process Definition and Management》，Springer, 2012）介绍了过程和产品的软件和系统工程观点。Glass（《Facts and Fallacies of Software Engineering》，Prentice-Hall, 2002）提出了一种保证软件工程过程的不加修饰且实用的观点。Brooks（《The Mythical Man-Month》，2d ed., Addison-Wesley, 1995）在他的书中虽然没有直接讲过程，但是他用一生的项目学识讲了和过程相关的每一个方面。

Firesmith 和 Henderson-Sellers（《The OPEN Process Framework: An introduction》，Addison-Wesley, 2001）为创建“灵活但有序的软件过程”提出了一个通用的模板，并讨论了过程属性和目的。Madachy（《Software Process Dynamics》，Wiley-IEEE, 2008）讨论了一种对软件过程中的相关技术和社会因素进行分析的建模技术。Sharpe 和 McDermott（《Workflow Modeling: Tools for Process Improvement and Application Development》，2nd ed., Artech House, 2008）介绍了为软件和商业过程建模的工具。

网上有大量关于软件工程和软件过程的信息，和软件过程有关的最新参考文献可以在 SEPA 网站 www.mhhe.com/pressman 找到。