基于 P-F 方法的软件过程建模的复用性*

周之英

(清华大学 计算机科学与技术系,北京 100084)

E-mail: zgzy-dcs@tsinghua.edu.cn

摘要: P-F 方法可以直观而精确地提供软件过程的复用机制.软件过程的复用问题如同软件本身的复用性,具有同样重要的意义.为了抽象过程复用机制,P-F 方法使用 3 层复用结构:过程模板、模式和元模式.过程模板代表由 P-F 方法描述的部分过程的可复用类.模式是模板的拓扑结构.最底层是元模式,代表最基本的模式,也是建成良结构过程/模板/模式的基本材料.利用元模式来构造软件过程可以避免病态的软件过程,并可以改进软件过程管理.形式化地定义了可复用结构部件间的操作.这些操作帮助实现软件过程的定义、复用或集成.软件过程的可复用特性在许多方面增强了 P-F 方法的优点:便于不同管理层次的交流;满足当前快速变化环境的变动要求;提供创建、分析、执行、控制、委任过程责任以及便于记录文档资料的工具.P-F 方法的抽象特性,便于构造针对一般过程的、由 P-F 引擎驱动的 P-F 虚拟机或 P-F 计算机器.实验系统正在进行中.

关键词: 软件过程;Petri 网;可复用性;模板;模式;元模式

中图法分类号: TP311 **文献标识码**: A

软件开发是以低价和在规定时期内获得高质量的软件产品为最终目的.我们大致可以把软件开发技术的研究分成两大方面:研究软件产品的结构和改进开发软件产品的过程.软件开发的工具与环境实际上也可归入工具的结构和工具所反映的开发方法这两方面.

有关软件产品结构的技术实际上是由软件在动态或静态条件下的各种不同抽象形式组成,包括从源代码结构、文件结构和数据结构,一直到子系统或部件间关系、体系结构等.这些成果已经充分应用到各种特定类型的软件系统当中,成为某种软件系统的标准结构,如操作系统和编译系统等.当然,一些通用的软件结构研究:如代码模块的性质、软件体系结构、设计模式等都大大强化了广义的软件可复用性.

在软件工业的早期,软件过程一词还不十分明朗.反复的编码-纠错的循环过程就是第 1 个软件过程.当软件系统变得大而复杂时,瀑布模型、螺旋模型、增式模型等都通过各种方式对软件过程进行改进,如便于工程管理、避免过大的风险、使用过去成功项目的成功管理经验等.由于目前的开发特点是强调适应快速变化环境的需求,因此,大型软件项目的管理层必须从目标管理转向过程管理.软件过程的标准定义是,在开发和维护信息系统中除软件产品结构以外的所有一系列的活动、方法和实践的总和.一般地说,软件过程和相应的软件产品结构是有密切联系的.但在一般意义下研究软件过程的改进对指导特定软件系统的软件过程也是十分重要的.

软件过程改进的基本原则是采用过去项目中成功的实践经验.因此,理解、记录和复用部分软件过程应该是软件过程改进研究的一个重要方向.研究软件过程复用问题如同软件本身的复用一样是非常重要的.

本文讨论 P-F 方法的复用特性.P-F 方法是我们正在进行的一项软件过程建模和软件过程改

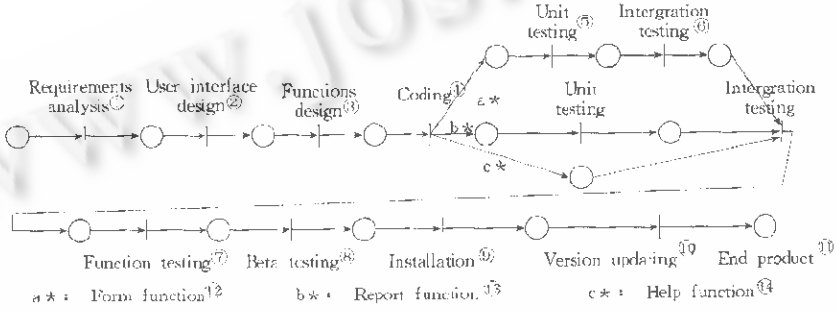
* 收稿日期: 2000-12-26; 修改日期: 2001-03-12

作者简介: 周之英(1939-),女,上海人,教授,主要研究领域为软件工程,软件过程建模,支持工具,搜索引擎.

进的方法论研究.本文第1节简单介绍P-F方法.第2节讨论P-F方法的3层抽象复用机制.第3节讨论在可复用对象上的关系和操作,以此组成P F方法的复用体系结构.最后是小结和进一步的研究方向.

1 P-F方法简介

开发大型复杂软件系统的关键之一是管理问题[1].正确管理的重要依据是对过程的可见性和在对软件过程了解的基础上进行选择、评估和决策.同软件产品的复用一样,软件过程的复用也可能帮助管理人员继承过去成功的辉煌.因此,软件过程改进的重要前提是提供软件过程的可见性和可复用性.我们提出一种以描述性为基础的方法——P F方法来帮助软件过程的改进.在P-F方法中,P代表Petri Net[2],F代表形式化的属性表PAB.P-F也可以理解为过程的形式化方向(P to F).



①需求分析,②用户界面设计,③功能设计,④编码,⑤单元测试,⑥集成测试,⑦功能测试,⑧β测试,⑨安装,⑩版本升级,⑪最终产品,⑫表格功能,⑬报告功能,⑭帮助功能.

(a) Petri net model
(a) Petri net模型

Activity name①	Activity description②(P)	Agent③ (A)	Beginning condition④(B)	End condition⑤ (E)	Input artifact⑥ (I)	Output artifact⑦(O)
Requirements analysis⑧	Visituser of TC department; Survey the system functions on IBM mainframe Write requirement specification⑨	A B	Setup project⑩	Review passed⑪	System prototype⑫	Requirement specification⑬
User interface design⑭	Design UI of system, including each query sub-system⑮	A B	Get familiar with MS Access⑯	UI design has included all content in requirement⑰	Requirement specification and user prototype⑱	Operating specification⑲

(b) PAB table
(b) PAB表

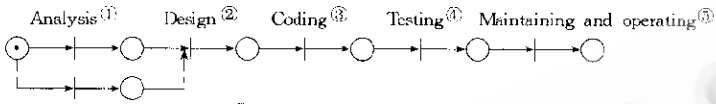
①活动名称,②活动描述,③代理,④起始条件,⑤结束条件,⑥输入产品,⑦产出产品,⑧需求分析,⑨访问技术合作部用户,调查IBM主机上的系统功能,编写需求说明书,⑩项目确立,⑪评审通过,⑫系统原型,⑬需求说明书,⑭用户界面设计,⑮设计系统的用户界面,包括对于系统的每个查询,⑯熟悉MS Access软件,⑰用户界面设计反映需求全部语境,⑱需求说明书和用户原型,⑲用户操作说明书.

Fig. 1 An example of concept model of software process in the real project (in an informal way)

图1 1个实际项目软件过程的概念模型例子(非形式化的)

P F方法由概念模型和实施模型组成.概念模型也称P-F模型,由Petri Net和PAB表组成.

直观而有坚实理论基础的 Petri Net 描述过程中的偏序异步并发活动. PAB 表可以非常规范地定义软件过程中的每一个活动. 图 1 给出了实际项目的软件过程的概念模型的例子. 按照管理的成熟程度,PAB 表中的各条目可以采用不同的形式化程度:从非形式化到完全形式化(可以直接自动化). 同时,PAB 表具有嵌套特性:每一个活动可以用 P-F 子模型来定义. 从而 P-F 方法可以用统一的方式来描述软件过程的集成和改变. 图 2 表示了用 P-F 方法表示过程的变化.



①分析,②设计,③编码,④测试,⑤维护运行,⑥修改分析.

Fig. 2 Process change represented by P-F method

图 2 用 P-F 方法表示过程的变化

实施模型通过继承过去成功项目的成果代表了大多数面向经验的知识. 实施模型包括所有收集、存储、复用和集成软件过程的工具. 实施模型也代表构造特定目的的概念模型的方法. 其工具是实施规则和复用机制. 实施规则有命名规则、分配地址规则、优化规则、并发和同步/异步规则等,通常也是从过去的成功项目中总结而来的知识. 实施规则在复用机制的帮助下,可以方便地定义新的软件过程的概念模型,使建模不必从零做起.

复用机制由一种 3 层抽象的复用结构——过程模板、过程模式和过程元模式组成.

P-F 方法与普通的 Petri Net 建模方法相比,通常情况下,Petri Net 是用于对异步并发过程的控制行为建模. 各种 Petri Net 的扩充(如带色 Petri Net,时间 Petri Net 等)都是使 Petri Net 模型更能反映现实世界[3]. 然而,所有的 Petri Net 扩展都仅仅是对系统的控制行为建模,而不能对系统的完整行为建模. 这也是为什么产生种类繁多的 Petri Net 扩展的根本原因. P-F 方法中的概念模型可以对具有并发异步过程的系统进行全面的建模:用 Petri Net 表示系统的并发行为,用 PAB 表表示过程中的活动定义. P-F 概念模型中 Petri Net 和 PAB 表的分离使整个概念模型呈现两种抽象层次:Petri Net 所代表的系统全局控制特征和 PAB 表中每个记录项所代表的局部功能的详细定义. 这样一来,P-F 方法给 Petri Net 的建模技术开辟了一条全新的方向:PAB 表归纳了所有现存的和未来的 Petri Net 扩展,甚至更多、更好. 我们可以方便地构造 P-F 虚拟机:一个由 Petri Net 引擎驱动、执行由 PAB 表中定义的活动的计算装置. 本文讨论 P-F 方法的复用机制,P F 虚拟机及 P-F 计算机器的特性和实现方案将在其他论文中加以阐述.

2 P-F 方法的 3 层复用机制

在 P-F 方法中存在 4 种对象:过程、模板、模式和元模式[4]. 除了过程以外,模板、模式元模式作为可复用的工具来支持建模工作.

模板定义. 过程或子过程的某些可复用部分的类.

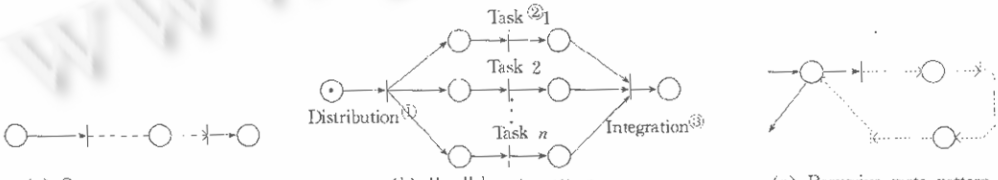
模板和过程一样也由 P-F 模型来描述,仅是在模板的 PAB 表中使用过程的 PAB 表中各条目的聚集. 换句话说,过程的某个部分可以是一个相应模板的实例. 例如,在过程中某节点的活动为"设计",其过程 PAB 表中的代理是"设计师小王";则在模板的 PAB 表中相应的代理则是"设计师",是设计师小王的聚集. 我们可以从 3 种途径获取模板:从过程中抽取;通过多个模板的组合;通过模式的发展.

模板的复用方法有 4 种:分部集成、层次嵌套、结构映射和库存剪裁. 其中分部集成和层次嵌套

由 P F 方法中的集成或抽取操作来实现.

从软件过程管理的观点来看,分部集成意味着在过程中包含不同管理经验的组合.层次嵌套意味着把管理责任委托给底层管理.高级经理在过程中以更大的粒度进行控制,以减小过程管理的复杂性.当然,我们可以重新对过程模型进行配置,如把多层转化为单层.这要求过程管理人员自己管理每一个细节.结构映射代表了在某些情况下问题结构应该与过程结构相一致的概念.当我们访问一个已经存在的过程模板库时,对模板库的成员进行剪裁应用则是常用的复用方法.模板库可以从公司内外的资源中获取.把以上 4 种复用方法组合使用,可以极大地增进创建过程的效率.

为了扩展 P-F 方法的可复用性,我们发展了模式与元模式的概念.在 P-F 方法中,模式是比模板更高层次的抽象,定义为模板的纯拓扑结构.实际上,模式把模板中的 PAB 表摈弃,只保持过程中各活动的控制流关系(Petri Net).只要对模式中每个活动进行具体定义,确定 PAB 表中的成分,我们就可以得到不同应用领域的过程模板.元模式定义为模式的最基本形式,是组成模式的基本模式,至今已经定义了 3 种元模式:顺序型元模式、并行型元模式和循环型元模式.这 3 种元模式可以对应于软件开发模型:瀑布模型、平行开发模型和螺旋模型.如图 3 所示.对元模式指定参数(元模式中节点数),得到 3 种基本类型的模式,同样,还可以应用上面提到的 4 种复用工具.



(a) Sequence meta-pattern
(a) 顺序型元模型

(b) Parallel meta-pattern
(b) 并行型元模式

(c) Recursive meta-pattern
(c) 循环型元模式

①分配,②任务,③集成.

Fig. 3 Three type of meta-patterns in P-F method
图3 在P-F方法中的3种元模型

良结构的模式是仅仅由元模式经过一定操作得到的模式.换句话说,良结构模式的管理风格是一系列的顺序活动、并行活动、循环活动和委托合同(嵌套功能相当于把责任委托给下层管理)的复合,这也代表了企业对业务工作和活动的正规管理类型.使用元模式可以避免病态的软件过程结构,加强和改进软件过程的管理.

3 P-F 方法中的关系和操作

为了便于过程的创建/修改,在 P-F 方法中定义了集成和抽取操作,这些操作的对象可以是一部分的过程、模板、模式和元模式.有 4 类操作:升级/降级操作(*Uop*/*Dop*)、活动合并/分解操作(*MTop*/*STop*)、产品状态合并/分解操作(*MSop*/*SSop*)、收缩/扩展(对嵌套)操作(*Sop*/*Eop*).升级操作 *Uop* 把一个模板/模式/元模式提升为过程/模板/模式.办法是把 PAB 表实例化/建立 PAB 表/指定元模式参数值.降级操作 *Dop* 是升级操作的逆操作,不必用于模式,不适用于元模式.其操作的实质是把过程 PAB 表中各项改为其聚集元素,或对模板来说要摒弃 PAB 表.收缩/扩展操作是针对嵌套特性而定义的操作.其意义十分清楚,这里不再作解释.

合并/分解操作可以是多元的.为简明起见,这里只对二元情况进行说明.活动合并操作 *MTop* 把两个过程(模板/模式)中的某项活动(即某项任务)合并为同一个,通过这一操作同时把两个过程(模板/模式)集成为一个新的过程(模板/模式).当然,这两个过程(模板/模式)也可以是同一个过程(模板/模式).在新的过程(模板/模式)中,该任务的具体定义将通过操作的参数来确定.参数规

定新任务的输入/输出产品,它们必须满足不变性约束.产品状态合并/分解操作(*MSop*/*SSop*)与活动合并/分解操作(*MTop*/*STop*)的性质类似.只是针对产品状态,而不是系统中的活动.操作同样应遵循不变性约束.

这里,我们给出操作的一些定义例子,由于篇幅所限,只给出顶层定义.用Z语言可以描述出完整的定义.

假定:P=过程集合;$p_1,\ldots,p_n\in P$;$s_i.p_k(/t_i.p_k)$是p_k中的一个产品状态(活动)

AT=模板集合;$at_1,\ldots,at_n\in AT$;$s_i.at_k(/t_i.at_k)$是at_k中的一个产品状态(活动)

AP=模式集合;$ap_1,\ldots,ap_n\in AP$;$s_i.ap_k(/t_i.ap_k)$是ap_k中的一个产品状态(活动)

APM=元模式集合

$S=p_k(/at_k/ap_k)$中的产品状态集合

$T=p_k(/at_k/ap_k)$中的活动集合

$B=p_k(/at_k)$中PAB表的集合

I=整数集

R=不变性约束集合

则
$$MTop:(P\times T)\times(P\times T)\times R\to P\mid(AT\times T)\times(AT\times T)\times R\to AT\mid(AP\times T)\times(AP\times T)\times R\to AP$$

说明:*MTop*是在过程间(或模板间或模式间)指定的活动上的操作,操作参数指定不变性约束(取决于操作的物理意义).操作结果也是一个过程,或一个模板,或一个模式.

$$MSop:(P\times S)\times(P\times S)\times R\to P\mid(AT\times S)\times(AT\times S)\times R\to AT\mid(AP\times S)\times(AP\times S)\times R\to AP$$

$$Uop:(AT\times B)\to P\mid(AP\times B)\to AT\mid APM\times I\to AP$$

$$Dop:P\to AT\mid AT\to AP$$

$$Eop:P\times(P\times T)\times R\to P\mid AT\times(AT\times T)\times R\to AT\mid AP\times(AP\times T)\times R\to AP$$

说明:*Eop*把某个过程(P)作为另一个过程中某项活动($P\times T$)所嵌套的子过程,当然,在操作中必须满足一定的约束关系,操作结果是一个新的过程(P),即将嵌套子过程展开到父过程中.我们可以看到,这些操作常常互为逆操作.如图4和图5所示,是P-F方法支持工具PSTOOL[5]中相应操作的一些界面.

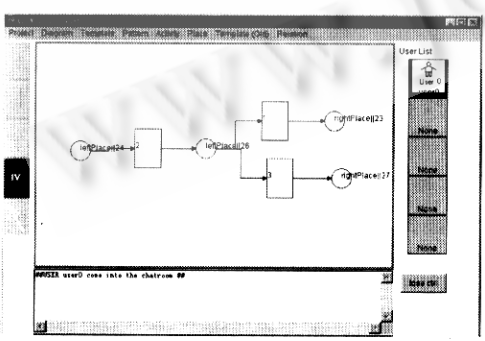


Fig. 4 The inteface before split operation

图4 在分解操作之前的界面

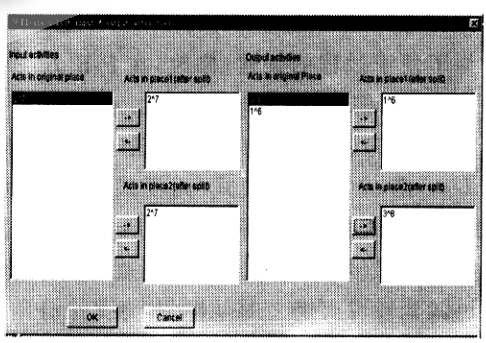


Fig. 5 The interface showing parameters for invariant rules

图5 用参数方式保证不变性约束的界面

图4显示了在分解操作以前的界面例子.为了确定分解操作,我们必须具体定义分解操作,使

其满足不变性约束.在 PSTOOL 中,用如图 5 所示形式的界面,约束用户来定义操作的结果过程中活动的相应输入/输出产品.图 6 显示了分解操作完成后的结果界面.我们可以用组合方式产生更高级的操作.所有的模板、模式和元模式加上以上操作组成 P-F 方法的完整的复用体系结构.这样一来,我们可以很方便地利用 P-F 方法的复用机制创建和修改过程模型.

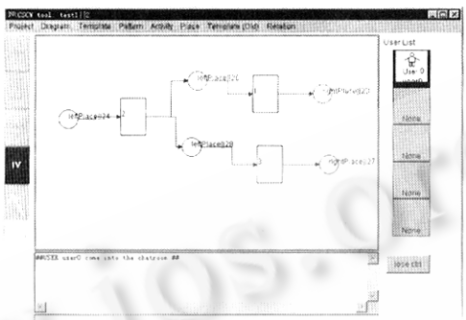


Fig. 6 The result interface of operations in PSTOOL

图 6 在 PSTOOL 中操作的结果界面

4 小 结

本文简明地介绍了 P-F 方法,讨论了 P-F 方法的复用体系结构.P-F 方法的复用特性使 P-F 方法的优点更加突出.我们可以归纳如下:

(1) 不同的高层和低层使用统一的描述方式,便于不同管理层次的沟通.

(2) Petri Net 的不确定性和循环型元模式提供了建模的灵活性.使 P-F 方法保持动态特性.更加适应当前的快速变动要求.

(3) 过程/模板/模式/元模式逐次抽象的层次结构提供了分析/控制/委托责任、过程执行和过程更动的有力工具.

(4) P-F 方法的扩展能力有利于不同类型的企业管理方式(从初级的/不成熟的到高级的/成熟的/规范化的).因此,P-F 方法可以成为一种无缝的过程连续改进工具.

(5) P-F 方法的描述能力可使其与其他软件过程改进方法结合起来,相辅相成.

(6) Petri Net 的严格的理论背景使 P-F 方法的自动工具的设计和实现很方便.

(7) 最重要的是,Petri Net 和 PAB 表的分离就是建模过程中全局性行为与细节定义的分离.开辟了建模技术的新方向.由于 P-F 方法的抽象性,如果我们把软件过程扩展到一般过程,主要的论点也同样成立.由 P-F 引擎驱动的 P-F 虚拟机或 P-F 计算机器可以作为现代计算的基础结构.

目前,我们正在进一步研究和完善 P-F 方法的理论基础,P-F 方法的工具原型 PSTOOL 也在不断发展之中.需要进一步发展 P-F 方法的实际应用,从而改进 P-F 方法是我们重点的努力方向.

References:

[1] Zhou, Zhi-ying. P-F methods for supporting software processes improving. In: Jin, Guo-fan, *et al* ed. Proceedings of the 5th National Joint Conference on Computer Application. Beijing: Electronic Industry Publishers, 1999. 4~80 (in Chinese).

[2] Toshifuni, T., Keishi, S., Shinji, K, *et al*. Improvement of software process by process description and benefit estimation. In: Proceedings of the ICSE'95. LOS Alamitos, CA: IEEE Press, 1995.

[3] Pree, W. Design Patterns for Object-Oriented Software Development. Titsworth: Addison-Wesley Publishing Company, 1995.

[4] Martin Naedele. Modeling and simulating functional and timing aspects of real-time systems by delegated execution. In: Proceedings of the 7th IEEE International Conference and Workshop on the Engineering of Computer Based Systems. Edinburgh, Scotland: ACM Press, 2000. 64.

[5] Zhou, Zhi-ying, Lu, Hai-peng. A Java-based CSCW tool for supporting software processes. In: Proceedings of the 7th IEEE International Conference and Workshop on the Engineering of Computer Based Systems. Edinburgh, Scotland: ACM Press, 2000. 368.

附中文参考文献:

[1] 周之英.支持软件过程改进的P-F方法.见:金国藩,等编,第5届全国计算机应用联合学术会议论文集.北京:电子工业出版社,1999.4~80.

Reusability Based on P-F Method for Software Process Modeling*

ZHOU Zhi-ying

(*Department of Computer Science and Technology, Tsinghua University, Beijing* 100084, *China*)

E-mail: zgzy-dcs@tsinghua.edu.cn

Abstract: P-F method provides the reusable mechanism of the software process with intuition and accurate. The reusability of software processes is very important feature like the reusability of software products. Three levels of reusable mechanism of P-F method for modeling the software process are the process template, the pattern and the meta-pattern. The process template represents the reusable class of process parts described by P-F method. The pattern is the topology structure of the template. The lowest reusable level is meta-pattern, which is basic style of pattern and is also part-style for constructing the well-structured processes/patterns. Using meta-patterns, it would avoid the ill structures of software processes and enhance the management of software processes. The operations among the template, pattern and meta-pattern are formally defined, which help to define, reuse and integrate software process. The reusability feature enhances the advantages of P-F method in many aspects: easing the communication among the different management layers; changing demands from recent rapid shift environment; providing the process tools to creation, analyze, execute, control, commit process responsibility; supplying the convenience of documentation. Because P-F method is quite abstract, it also could be applied to general processes, and implemented by a P-F virtue machine or P-F computing machine driven by the P-F engine. The prototype is undergoing.

Key words: software processes; Petri net; reusability; template; pattern; meta-pattern

* Received December 26, 2000; accepted March 12, 2001