

# 面向对象方法的 COSMIC-FFP 功能规模度量

计春雷<sup>1,2</sup> 闫顺成<sup>1</sup> 宋国新<sup>1</sup>

(华东理工大学计算机科学与工程系 上海 200237)<sup>1</sup> (上海电机学院电子信息学院 上海 200240)<sup>2</sup>

**摘 要** 现有的 4 种符合 ISO 标准的 FSM 方法均不能考虑到对象的交互性和对象的行为,无法正确地度量面向对象系统的功能规模。在分析面向对象方法的软件开发过程的基础上,结合面向对象系统的特点,基于 COSMIC-FFP,提出了一种面向对象方法的全功能点度量方法,给出了该方法的映射规则和度量规则,并结合实例分析了其应用过程,为正确度量面向对象系统的功能规模提供了一条有效的途径。

**关键词** 面向对象方法,功能规模度量,COSMIC-FFP,面向对象方法的全功能点,映射

**中图法分类号** TP311 **文献标识码** A

## COSMIC-FFP Function Size Measurement for Object-oriented Methods

Ji Chun-lei<sup>1,2</sup> YAN Shun-cheng<sup>1</sup> SONG Guo-xin<sup>1</sup>

(Department of Computer Science and Engineering, East China University of Science and Technology, Shanghai 200237, China)<sup>1</sup>

(School of Electronic Information, Shanghai Dianji University, Shanghai 200240, China)<sup>2</sup>

**Abstract** The four existing Functional Size Measurement(FSM) methods that meet the ISO standards can not take the interactions and behaviors of objects into consideration and can not correctly measure the functional size of object-oriented systems. So, based on the analysis of software development process of object-oriented methods, according to the characteristic of object-oriented systems, this paper presented an Object-Oriented Full Function Point method based on COSMIC-FFP, and gave the mapping rules, measurement rules and an example of the application process of the approach. This method provided an effective way for the correct measure of the function size of object-oriented systems.

**Keywords** Object-oriented method, Functional size measure(FSM), COSMIC-FFP, Object-oriented full function point(OOFFP), Mapping

随着计算机技术的日新月异,软件规模的扩张具有令人叹为观止的速度,人们在软件开发过程中越来越深刻地认识到软件规模度量的重要性。功能规模度量(Functional Size Measurement, FSM)方法从用户角度和功能角度来度量软件功能规模,与程序设计语言无关,而且在项目开发的初期可利用需求分析模型进行功能点的估算<sup>[1,2]</sup>。

面向对象方法(Object-Oriented Method, OO 方法)被誉为“研究高技术的好方法”<sup>[3]</sup>,正引起全世界越来越广泛的关注和高度的重视。而现有的 4 种符合 ISO 标准的 FSM 方法不能考虑到对象的交互性和对象的行为<sup>[4]</sup>,无法完全正确地度量出面向对象系统的功能规模。

在分析面向对象方法开发软件过程的基础上,结合 COSMIC-FFP 的特点,提出了一种适合度量面向对象系统的、面向对象方法的全功能点度量方法(Object-Oriented method Full Function Point, OOFFP),为在软件开发初期正确度量出面向对象系统的功能规模提供了一条有效的途径。

## 1 COSMIC-FFP 度量方法简介

COSMIC-FFP<sup>[3,5]</sup>度量方法是由加拿大 St Pierre 等人于

1997 年提出的,主要用来度量实时系统、嵌入式系统和商业应用软件的功能规模。

### 1.1 COSMIC-FFP 度量要素

COSMIC-FFP 方法认为软件的功能规模可以通过有效“数据移动”的个数来度量,其功能要素就是数据移动类型。COSMIC-FFP 度量模型定义了 4 种类型的数据移动:输入(Entry)、输出(Exit)、读取(Read)和写入(Write)。一个有效的“数据移动”记为一个 COSMIC 功能点(Cosmic Function Point, CFP)。COSMIC-FFP 方法中涉及到的关键概念有用户、层、边界、功能过程和数据组<sup>[5]</sup>。

### 1.2 COSMIC-FFP 度量过程

COSMIC-FFP 方法的度量过程分为两个阶段<sup>[5]</sup>:第一个阶段是映射阶段,其目的是将软件的用户功能需求映射到 COSMIC-FFP 元模型中的元素,如功能处理、数据组、数据属性;第二个阶段是度量阶段,其目的是将功能处理分解为数据移动并计算出功能规模,最终得到软件规模大小。

## 2 面向对象方法的软件开发过程简介

面向对象方法(OO 方法)是一种把面向对象的思想应用

到稿日期:2010-06-08 返修日期:2010-09-12 本文受国家自然科学基金项目(60773094)资助。

计春雷(1964—),男,博士生,副教授,主要研究方向为软件度量技术、嵌入式技术, E-mail: jiclei@sina.com; 闫顺成(1983—),男,博士生,主要研究方向为软件工程; 宋国新(1945—),男,硕士,教授,博士生导师,主要研究方向为软件工程、软件自动化、知识工程。



R8:消息类型的每个蕴含了恢复属性值的条件都识别为一个 Read 数据移动类型。

R9:用例的详尽规范中定义每个前置条件都识别为一个 Read 数据移动类型。

R10:扩展关系的每个条件都识别为一个 Read 数据移动类型。

R11:每个数据一致性约束都识别为一个 Read 数据移动类型。

R12:每个具有《service/new》,《service/destroy》或《service/update》和《connect》版型的消息都识别为一个 Write 数据移动类型。

R13:每个具有版型《signal》和输出值的消息都识别为一个 Exit 数据移动类型。

### 3.1.2 消除冗余数据移动阶段的映射规则

COSMIC-FFP 中的术语“de-duplication”意思是消除冗余数据移动的过程<sup>[4,5]</sup>。下面给出一些规则可以防止在度量阶段度量重复的数据移动:

R14:接受的对象相同、移动的数据组也相同并且版型也相同的所有消息仅识别为一个数据移动。

R15:在顺序图的一个模块中识别的数据移动集合只能计算一次;在顺序图的一个模块中的 include 消息只能计算一次。

R16:用例范围里所有的确认和错误消息作为一种 Exit 数据移动仅识别一次。

### 3.2 定义度量规则

为了根据面向对象方法的需求模型计算出面向对象软件的功能规模的量化值,本文采用了 COSMIC-FFP 度量函数<sup>[4,5]</sup>,并提出了一些度量规则来得到度量结果。

#### (1)应用 COSMIC-FFP 度量函数

对每个功能过程中识别的每个数据移动应用 COSMIC-FFP 的度量函数。该函数描述为:每个识别了的数据移动产生一个 CPF(Cosmic Functional Point)规模值。

$$f(x)=1Cfp \forall x \in P/P=\{entry, exit, read, write\}$$

#### (2)度量功能过程的功能规模

应用聚集函数将功能过程中所有被识别了的数据移动的功能规模添加到该功能过程的度量结果中。此聚集函数由下面规则表示:

R17:一个用例的功能规模等于该用例中所有识别的数据移动的总量。

该规则可由下式表示:

$$Size(UseCase)=\sum_{i=1}^n f(x)_i$$

然而,由于用例图之间存在的关系,因此定义了两个附加规则。为了度量由一个或多个次级用例(SUC)所扩展的用例的功能规模,聚集函数可由如下规则来说明:

R18:一个由次级用例集合扩展的基本用例(BUC)的功能规模等于每个次级用例中识别的数据移动和该基本用例中的数据移动的数量之和。

相似地,对于一个 INCLUDE 关系,一个基本用例的功能规模解释如下:

R19:一个包含了其它次级用例的基本用例的功能规模等于每个包含的用例中识别的数据移动和该基本用例中的数据移动的数量之和。

这两个规则可以用下面的等式来表示:

$$Size(BUC)=\sum_{i=1}^n Size(SUC_i)+Size_p(BUC)$$

表达式  $Size_p$  是指基本用例的部分规模,它等于所有在这个用例中识别的数据移动的数量。

#### (3)度量软件层的功能规模

应用聚集函数将软件层中所有被识别为功能过程的主用例(PUC)集合的功能规模添加到该软件层的度量结果中。在这个层面上不再考虑次级用例以避免度量中的冗余。因此有下面的规则:

R20:一个软件层的功能规模等于所有主用例的功能规模之和。

该规则可由下面的等式表示:

$$Size(Layer_1)=\sum_{i=1}^n Size(PUC_i)$$

## 4 OOFFP 度量应用实例分析

一个功能规模度量方法的应用是在一个复杂、抽象的环境下进行的,包括一个度量模型与软件模型间的映射阶段和一个将度量规则实例化派生出映射模型的度量阶段<sup>[11]</sup>。按照 Jacquet 和 Abran 的度量理论<sup>[7]</sup>,应用一个度量方法需要分 3 步进行:软件文档整理、软件模型的建立和度量规则的应用。

下面以某图书信息管理系统为案例来阐述 OOFFP 方法的应用。

### 4.1 软件文档整理

图书信息管理系统的主要功能包括基本业务(借阅图书和归还图书)处理、信息查询和系统维护等。

该系统的功能需求树(FRT)是基于问题陈述的定义来建立的,它包括 3 个功能组:借阅者服务、图书管理员管理、系统管理员管理。图 3 为借阅者服务的用例图,它表达了图书信息管理系统能够为借阅者提供的服务功能。

为了度量该借阅者服务功能的功能规模,对此用例图中的“查找图书”、“查询个人信息”和“预定图书”用例需要用顺序图来表示。由于“借阅图书”、“归还图书”和“缴纳罚金”用例主要是由图书管理员完成的,因此这些不在此处考虑。

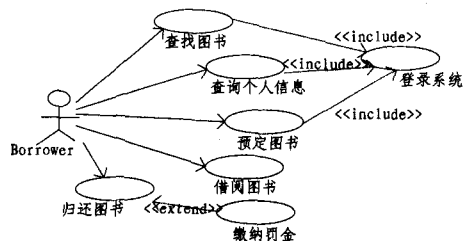


图3 借阅者服务

### 4.2 软件模型的建立

应用 OOFFP 方法的映射规则建立图书信息管理系统的软件模型。

首先确定度量范围由“借阅者服务”功能组中指定的功能组成。由于指定的功能需求属于同一个的抽象层,因此识别为一个单独的软件层。然后应用如下的映射规则来建立相应的软件模型:

用例图中 Borrower 为参与者,由规则 R1 识别出 Borrower 为用户。

由规则 R2 识别出用例图中 Borrower 与用例之间为软件边界。

由规则 R3 识别出主用例“查找图书”、“查询个人信息”和“预定图书”为功能过程,次级用例“登录系统”也为功能过程。

限于篇幅,本文使用“预定图书”功能过程为例子阐述数据移动的识别,该功能过程的顺序图如图 4 所示。

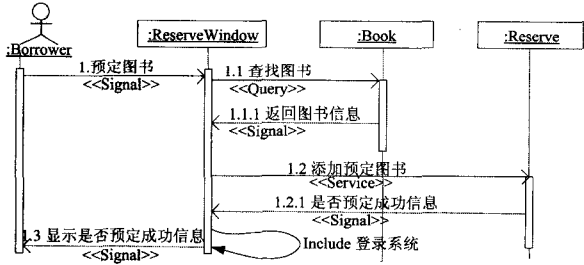


图 4 预定图书顺序图

由规则 R4 识别出顺序图(图 4)中的类 ReserveWindow, Book 和 Reserve 为数据组,由规则 R1 确定的用户(Borrower)也为数据组。

该功能过程起始于用户对数据的引入,这由消息“预定图书”表示。为此应用规则 R6 识别出一个 Entry 类型的数据移动。数据引入以后,借阅者用“查找图书”消息来请求查找该图书,应用规则 R7 识别出该消息是一个单独的 Read 类型的数据移动。不管图书是否存在,系统用“返回图书信息”的消息返回,因此由规则 R13 可知该消息产生了一个 Exit 类型的数据移动。如果图书存在,系统通过使用“添加预定图书”消息将图书信息加入预定中,应用规则 R12 识别出这个消息是 Write 类型的数据移动;同时系统通过“是否预定成功信息”的消息显示预定是否成功,应用规则 R13 识别出该消息是一个 Exit 类型的数据移动。最后,系统通过“显示是否预定成功信息”的消息向借阅者返回该书是否预定成功的信息,应用规则 R13 识别出该消息也是一个 Exit 类型的数据移动。“预定图书”功能过程中的数据移动识别结果如表 1 所列。

表 1 “预定图书”功能过程中的数据移动

| 消息标号  | 消息名称       | 数据移动类别 | 应用规则 |
|-------|------------|--------|------|
| 1     | 预定图书       | Entry  | R6   |
| 1.1   | 查找图书       | Read   | R7   |
| 1.1.1 | 返回图书信息     | Exit   | R13  |
| 1.2   | 添加预定图书     | Write  | R12  |
| 1.2.1 | 是否预定成功信息   | Exit   | R13  |
| 1.3   | 显示是否预定成功信息 | Exit   | R13  |

在预定图书过程中,借阅者也调用了“登录系统”用例。由于该用例在“查找图书”和“查询个人信息”功能过程中也需使用,因此由规则 R14 可知该用例在度量时仅需考虑一次。

同理,为其余已识别的功能过程识别出数据移动以及消除冗余的数据移动。

### 4.3 度量规则的应用

为了得到图书信息管理系统的功能规模值,本文应用了前面已定义的度量规则。

首先,为前面每一个步骤中识别的每个数据移动赋一个 1CFP 规模值。

然后,应用度量规则完成如下的规模度量:

(1)“预定图书”用例的数据移动识别结果如表 1 所列,共有 6 个数据移动。应用规则 R17,得到“预定图书”用例的部分规模为 6CFP;同样地,“登录系统”用例中数据移动为 2 个,得到“登录系统”用例的功能规模为 2CFP。

(2)应用规则 R19,将“预定图书”主用例的部分规模与“登录系统”的规模相加,得到“预定图书”用例的功能规模值为 8CFP。同样地,按照上面介绍的相同步骤,消除冗余后,可计算出“查找图书”用例的功能规模值为 5CFP,“查询个人信息”用例的功能规模值为 4CFP。

最后,计算出图书信息管理系统的“借阅者服务”模块的功能规模为 17CFP。

**结束语** 本文结合面向对象系统的开发特点和 COSMIC-FFP 的度量要素,提出了面向对象方法的 COSMIC-FFP 度量方法(OOFFP)。OOFFP 描述了从软件需求说明度量软件功能规模的一套规则,实现了从面向对象方法到 COSMIC-FFP 方法的概念上的映射和度量规则上的映射。OOFFP 方法的主要贡献是面向对象系统的功能规模可以在开发过程的早期阶段度量。其优点是从用户功能需求出发,被度量软件的文档可以直接作为度量过程的输入,度量过程步骤清晰,度量人员进行度量的难度不大。进一步的研究工作有:(1)实现用形式化的方法描述 OOFFP 的映射与度量规则;(2)研究软件的自动化度量的解决方案,设计并实现 OOFFP 的自动化度量系统。

### 参考文献

- [1] Low C G, Jeffery R. Function point in the estimation and evaluation of software process [J]. IEEE Trans. on Software Engineering, 1990, 16(1): 64-71
- [2] ISO/IEC 14143-1:1998. Information Technology-Software measurement- Functional Size Measurement-Part1: Definition of Concepts [S]. 1998
- [3] 胡圣明,褚华. 软件设计师教程(第三版)[M]. 北京:清华大学出版社,2009:517-548
- [4] 顾勋梅,邵志清. 面向对象方法的功能点度量[J]. 计算机工程与应用, 2008, 44(36): 90-94
- [5] ISO/IEC 19761:2003. COSMIC Full Function Points Measurement Manual v. 2. 2 [S]. 2003
- [6] Pastor O, Gomez J, Insfran E, et al. The OO-Method approach for information systems modeling: from object-oriented conceptual modeling to automated programming[J]. Information Systems, 2001, 26(7): 507-534
- [7] Jacquet J P, Abran A. From Software Metrics to Software Measurement Methods: A Process Model [C]// Proceedings of the Third International Symposium and Forum on Software Engineering Standards ( ISESS ' 97 ). IEEE. Walnut Creek ( CA ). 1997: 1-12
- [8] Abrahao S, Poels G, Pastor O. A functional size measurement method for object-oriented conceptual schemas: design and evaluation issues [J]. Software & System Modeling, 2006, 5(1): 48-71
- [9] Poels G. Functional size measurement of multi-layer object-oriented conceptual models [C]// Proc. 9th International Object-Oriented Information Systems Conference. Geneva, Switzerland. 2003: 334-345
- [10] Larman C. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development (3rd Edition) [M]. Prentice Hall PTR, 2004: 123-145
- [11] Desharnais J, Abran A. Applying functional measurement method; Cognitive issues [C]// Current Trends in Software Measurement. Shaker Verlag, Germany, 2001: 26-50