



Springer

程序设计 方法学

Programming Methodology

[美] Annabelle McIver Carroll Morgan 著

田玉敏 等译



中国水利水电出版社
www.waterpub.com.cn

TP311.11/45

2007

程序设计方法学

[美] Annabelle McIver Carroll Morgan 著

田玉敏 等译

中国水利水电出版社

内 容 提 要

本书重点介绍新的和正在出现的构造先进应用程序的技术,解决软件设计人员开发高度复杂的应用程序时必须面临的问题。本书的主要内容包括:模型与正确性、程序设计技术以及应用和自动机理论。同时,给出了几个来自不同领域的应用实例,例如安全、电话和电路设计等。

本书是先进的计算机软件理论和实践方面的权威性指导书,适合作为计算机科学家、程序设计人员和软件工程师的参考书。

Translation from the English language edition: *Programming Methodology* edited by Annabelle McIver and Carroll Morgan.

Copyright © 2003 Springer New York, Inc.

All Rights Reserved.

北京市版权局著作权合同登记号:图字 1-2004-1433

图书在版编目(CIP)数据

程序设计方法学 / (美)麦基弗 (McIver, A.), (美)摩根 (Morgan, C.) 著; 田玉敏等译. —北京:中国水利水电出版社, 2007

书名原文: *Programming Methodology*

ISBN 978-7-5084-4368-3

I. 程… II. ①麦…②摩…③田… III. 程序设计—方法
IV. TP311.11

中国版本图书馆 CIP 数据核字 (2007) 第 155639 号

书 名	程序设计方法学
作 者	[美] Annabelle McIver Carroll Morgan 著
译 者	田玉敏 等译
出版 发行	中国水利水电出版社 (北京市三里河路 6 号 100044) 网址: www.waterpub.com.cn E-mail: mchannel@263.net (万水) sales@waterpub.com.cn 电话: (010) 63202266 (总机)、68331835 (营销中心)、82562819 (万水)
经 售	全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	北京蓝空印刷厂
规 格	787mm×1092mm 16 开本 22 印张 533 千字
版 次	2007 年 10 月第 1 版 2007 年 10 月第 1 次印刷
印 数	0001—4000 册
定 价	48.00 元

凡购买我社图书,如有缺页、倒页、脱页的,本社营销中心负责调换

版权所有·侵权必究

撰稿者名单

WG2.3 工作小组的成员名单:

Professor Dr.R.-J.Back

Dr.R.M.Balzer

Dines Bjørner

Professor Dr.M.Broy

Dr.Ernie Cohen

Dr.P.Cousot

Professor Dr.E.W.Dijkstra

Professor D.Gries

Professor I.J.Hayes

Professor E.C.R.Hehner

Professor P.Henderson

Professor C.A.R.Hoare

Dr.J.J. Horning

Professor Daniel N.Jackson

Mr.M.A.Jackson

Professor C.B.Jones

Dr.B.W.Lampson

Dr.K.Rustan M.Leino

Dr.M.D.McIlroy

Dr.Annabelle McIver

Dr.W.McKeeman

Dr.K.McMillan

Professor J.Misra

Dr.Carroll Morgan (vice-chairman)

Dr.Greg Nelson

Professor Benjamin Pierce

Dr.J.R.Rao

Professor J.C.Reynolds

Dr.D.T.Ross

Professor F.B.Schneider

Dr.N.Shankar (chairman)

Professor M.Sintzoff

Dr.Joakim von Wright

Dr.Pamela Zave

撰稿成员名单:

Dines Bjørner

Department of Information

Technology

Technical University of Denmark

DTU-Building 344

DK-2800 Lyngby

Denmark

Ernie Cohen

ernie.cohen@home.com

Ian Hayes

School of Information Technology and

Electrical Engineering

Ralph Back

Abo Akademi University

Department of Computer Science

Lemminkäinenkatu 14

SF-20520 Turku

Finland

Manfred Broy

Institut für Informatik

Technische Universität München

D-80290 München

Germany

Rustan Leino

Compaq SRC

130 Lytton Avenue

The University of Queensland
Queensland,4072
Australia

Eric C.R.Hehner
Pratt Building,Room PT398
University of Toronto
6 King's College Road
Toronto,Ontario M5S 3H5
Canada

Peter Henderson
Department of Electronics and
Computer Science
University of Southampton
Southampton,SO17 1BJ
United Kingdom

Tony Hoare
Microsoft Research
Cambridge, CB3 0FB
United Kingdom

Daniel Jackson
Lab.for Computer Science
200 Technology Square
Cambridge, Massachusetts 02139
USA

Michael Jackson
jacksonma@acm.org

Benjamin Pierce
University of Pennsylvania
Department of Computer and
Information Science
200 South 33rd Street
Philadelphia, Pennsylvania
19104-6389
USA

Palo Alto, California 94301
USA

Annabelle McIver
The Department of Computing
Macquarie University
Sydney, 2109
Australia

Jayadev Misra
Department of Computer Sciences
University of Texas at Austin
Austin, Texas 78712-1188
USA

Carroll Morgan
Department of Computer Science and
Engineering
The University of New South Wales
Sydney, 2052
Australia

Greg Nelson
Compaq SRC
130 Lytton Avenue
Palo Alto, California 94301
USA

C.B.Jones
Department of Computing Science
University of Newcastle
Newcastle-upon-Tyne, NE1 7RU
United Kingdom

John C.Reynolds
Computer Science Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, Pennsylvania 15123-3890
USA

Josyula Rao
IBM Research
T.J.Watson Research Center
P.O.Box 218
Yorktown Heights, New York 10598
USA

Natarajan Shankar
SRI International
MS EL256
333 Ravenswood Avenue
Menlo Park, California 94025-3493
USA

其他撰稿者名单:

Suresh Chari
IBM Research
T.J.Watson Research Center
P.O.Box 714
Yorktown Heights, New York 10598
USA

Charanjit S.Jutla
IBM Research
T.J.Watson Research Center
P.O.Box 714
Yorktown Heights, New York 10598
USA

Pankaj Rohatji
IBM Research
T.J.Watson Research Center
P.O.Box 714
Yorktown Heights, New York 10598
USA

Joakim von Wright
Abo Akademi University
Department of Computer Science
Lemminkainenkatu 14
SF-20520 Turku
Finland

Pamela Zave
180 Park Avenue
P.O.Box 971
Florham Park, New Jersey 07932-2971
USA

He Jifeng
International Institute of
Software Technology
United Nations University
P.O.Box 3058
Macau

Theodore Norvell
Electrical and Computer Engineering
Faculty of Engineering
Memorial University of Newfoundland
St.John's, NF A1B 3X5
Canada

Richard F.Paige
Department of Computer Science
York University
4700 Keele Street
Toronto, Ontario M5J 1P3
Canada

译者序

程序设计方法学是 20 世纪 60 年代末 70 年代初形成和发展起来的计算机科学领域中的一个新兴学科,随着计算机硬件技术的飞速进步,这一学科也得到了迅速的发展,取得了许多令人鼓舞的成果。

本书重点介绍已经出现和正在出现的先进的应用程序构造技术,从实际应用出发,论述在开发高度复杂的应用程序时必须面临的问题的解决方法。本书中的论文主要来自国际信息联合会 2.3 工作小组的一流学者和企业家的研究成果。

本书共分三部分,每一部分涉及程序设计方法学的一个专题。第一部分主要论述描述、建模和分析复杂的实际应用系统的方法;第二部分集中论述专用的程序设计技术——“程序员的工具包”;第三部分详细论述了一些应用,以及自动机理论和并发系统在可靠性设计方面的一些难题。

总之,本书是程序设计方法学方面的一些新的关键基础课题的最新技术水平的综述,汇集了国际上一流的计算机科学家、程序设计人员和软件系统设计人员的宝贵研究资料。本书作为先进的软件理论和实践方面的权威性指导书,适合程序设计人员、开发人员、计算机科学家和软件工程师参考。

本书内容涉及面广,理论性强,其中有些内容并不是译者的专长,虽然在翻译过程中译者尽可能地查阅了一些资料,但不完善的地方在所难免,欢迎广大读者和同行批评指正。

译者

2007 年 4 月于西安电子科技大学

前 言

20 世纪后半叶，计算机的计算能力得到了极大的提高。现今的计算机比曾经的计算机速度快得多得多，它们内存更大，日常能够与世界各地的远程计算机进行通信——而且能够安放在一张桌子上。但是，尽管取得了这样显著的进步，现今的用户仍迫切期望能够进一步将技术发挥到极致。随着硬件工程师建造的计算机功能日益强大，软件为了赶上硬件的发展也必然变得复杂得多。

中规模和大规模程序设计项目需要一个团队在可接受的时间期限内把所有的东西集成在一起。程序员如何理解自己的工作、如何将自己的工作与其他同事的工作集成在一起以达到总体指标等问题是关注的重点。如果没有这样的理解，了解现今的计算硬件的商业潜能实际上是不可能的。

程序设计能够赶上硬件的巨大进展要归功于程序设计、构造以及组织原理的相应的巨大进展。这些方法和原理的功效不言而喻——计算机技术遍及各方面——但更重要的是它们已经开始反过来影响硬件的设计。对这样的方法的研究称为程序设计方法学，其研究课题包括系统模型、域模型、并发、面向对象、程序规范与验证等。

这些是本论文集的主题。

程序设计方法学

现今的大多数系统的目标是安全、可靠、易于使用和实时。为了达到这些目标，程序员需要合适的工具。这些工具在此环境中是“基于智能的”，并且包含有助于以研究人员能够理解、计算机能够解释的方式组织和表示复杂问题的技术。

降低复杂性（或者只要可能，至少隐藏复杂性）的期望一直是发明设计原理和方法的动力，它们中的许多现在已经集成在普遍使用的程序设计语言和（自动）程序开发工具中。例如，类型语言可以帮助进行错误检测，而面向对象语言和数据抽象（在 Java 中二者都存在）支持程序修改、接口级程序设计和易读性。同时，随着并发语言以及形式化工具（包括验证中所使用的模型检验和证明帮助）的引入，并发也已经开始非常流行。

这些工具中的许多核心有深刻的理论背景——“断言”和“程序不变量”依赖于程序设计逻辑理论；而规范和精化技术的程序语义是以它们为基础的。本论文集集中的短文集中论述构造先进应用程序的新方法和正在涌现的方法；它们要解决软件开发人员面临的问题，提出实际解决方案和它们的理论基础。

汇编有关本主题论文成一本书的思想是在国际信息联合会 2.3 工作小组（IFIP）的技术研讨会上出现的。

工作小组 2.3

IFIP 的工作小组定期开会讨论新的思想——他们自己的和其他人的——评价和促进计算系统的许多方面的发展趋势。虽然他们常常提出特殊的课程和召开会议，但他们的官方作品随

小组的不同而差别很大，并且与当前成员的传统和风格关系极大。

术语“程序设计方法学”是由 WG 2.3 的成员之一提出的一个新词语，在小组内流行已将近 30 年，其成员对上述提到的许多课题做出了贡献；的确，现在程序设计方法学方面许多活跃的研究领域是在 WG2.3 会议上曾提出和讨论的。

关于本论文集

本卷论述的是该小组的第二官方出版物。我们的目的是收集能够吸引在大学或企业工作的学生和专业人员的资料。的确，我们希望本论文集能够作为程序设计方法学方面前沿研究活动的参考，并且能够指导程序设计方法学方面的前沿研究活动。

课题的内容能够反映当前成员的兴趣，能够解决与实际工作中高度复杂的应用程序的当前需求有关的具体问题。许多短文含有新的材料，能够强调特定的理论进展，而其他的一些目的是综述或评价某一具体领域的发展，或概述要进一步研究的发人深思的问题。

本书结构

本书包含三部分，每一部分涉及程序设计方法学的一个专题。每一部分又可以进一步分成几小部分，每一小部分汇集重点讨论这一部分主题内的某一具体课题的短文。每一小部分开始时的简短引言可以为之后的详细论述做准备。

系统可能很复杂，因为它们可能分布在网络上，或者它们是时间关键型或并发的系统——本书的第一部分主要论述描述、建模和分析这样的系统的方法，第二部分集中论述专用的程序设计技术——程序员的工具包，最后一部分详细说明一些课题的应用，包括安全性和电话学。

致谢

可以说，没有这些文章的作者的创造性工作，就没有本书。我们特别感谢 Natarajan Shankar (WG2.3 的主席)，是他最先提议进行这一项目的。此外，也要特别感谢 David Gries，感谢他在本书实现的过程中所给予的帮助。

Annabelle McIver

Carroll Morgan

澳大利亚，悉尼，2002

目 录

撰稿者名单

译者序

前言

第一部分 模型与正确性

Section A 并行与交互	1
第 1 章 需要：并行性的组构方法	3
1.1 组构性	3
1.2 并发性的本质是干扰	4
1.3 推理干扰	5
1.4 关于假设/承诺推理的一些问题	7
1.5 寄生变量的作用	7
1.6 粒度所关心的事情	7
1.7 抽象的原子性及其精化	8
1.8 结论	8
1.9 致谢	8
参考文献	9
第 2 章 用契约强制行为	11
2.1 引言	11
2.2 契约	12
2.2.1 状态与状态变化	12
2.2.2 契约	13
2.2.3 操作语义	14
2.2.4 契约举例	15
2.2.5 行动系统	16
2.2.6 行动系统举例	17
2.3 利用契约达到目标	18
2.3.1 最弱的前置条件	18
2.3.2 正确性与成功策略	20
2.3.3 契约的精化	22
2.4 强制行为属性	23
2.4.1 分析行为	23
2.4.2 构造解释程序	24

2.4.3 其他暂态属性	25
2.5 分析行动系统的行为	27
2.5.1 行动系统的分类	27
2.5.2 分析行为	28
2.6 验证强制	30
2.6.1 谓词级正确性条件	30
2.6.2 基于不变量的方法	31
2.6.3 示范方法	32
2.6.4 例子系统中的强制性	33
2.7 结论及相关工作	36
参考文献	36
Section B 异步逻辑方法	38
第3章 异步进展	40
3.1 引言	40
3.2 程序	41
3.3 达成	43
3.4 退耦	44
3.5 举例——松耦合程序	45
3.6 异步安全	46
3.7 警告	47
3.8 结论	48
3.9 致谢	48
参考文献	48
第4章 并发面向对象程序简化定理	50
4.1 引言	50
4.2 Seuss 程序设计符号	51
4.2.1 Seuss 语法	52
4.2.2 Seuss 语义(可选)	55
4.3 Seuss 程序模型	56
4.4 对程序的限制	58
4.4.1 方框上的偏序	58
4.4.2 把过程看作关系	59
4.4.3 方框条件	60
4.5 兼容性	61
4.5.1 兼容性举例	61
4.5.2 兼容过程的半交换性	62
4.6 简化定理的证明	64
4.6.1 松执行	64
4.6.2 简化方案	65

4.7 结束语	66
4.8 致谢	67
参考文献	67
Section C 系统与实时性	68
第 5 章 抽象时间	70
5.1 引言	70
5.2 流	70
5.2.1 数学基础: 流	71
5.2.2 建时间模型	71
5.3 把组件看作流函数	73
5.4 时间抽象	74
5.4.1 抽象的一般概念	74
5.4.2 抽象时间	75
5.5 结论	76
5.5.1 文献中的时间模型	76
5.5.2 结语	77
5.6 致谢	77
参考文献	78
第 6 章 实时精化的谓词语义	80
6.1 背景知识	80
6.2 语言和语义	81
6.2.1 实时规范命令	83
6.2.2 简单实时命令	86
6.2.3 顺序组合	86
6.2.4 不确定选择、条件和挑选	88
6.2.5 局部和辅助变量	90
6.3 一个例子	91
6.4 循环	93
6.5 定时约束分析	96
6.6 结论	97
6.7 致谢	98
参考文献	98
Section D 规定复杂的行为	100
第 7 章 系统描述的方面	102
7.1 引言	102
7.2 符号处理	102
7.3 机器和领域	104
7.3.1 规范接口	104
7.3.2 需求接口	104

7.3.3	系统描述	104
7.3.4	有目的的描述	105
7.3.5	为什么需要单独的描述	106
7.4	描述实际领域	107
7.4.1	标记	107
7.4.2	使用定义	108
7.4.3	区分定义和描述	109
7.5	描述与模型	109
7.5.1	电梯模型	110
7.5.2	模型的关系	111
7.5.3	实际模型	112
7.5.4	描述模型和所建模的域	112
7.6	问题分解与描述结构	113
7.6.1	检查子问题	113
7.6.2	诸多的矛盾	114
7.6.3	身份关注点	114
7.6.4	病人监测	115
7.7	软件开发的前景	115
7.7.1	无意义的规范	116
7.7.2	离散复杂性	116
7.8	致谢	117
	参考文献	117
第 8 章	建立动态系统的体系结构的模型	118
8.1	引言	118
8.1.1	动态系统	118
8.1.2	连续变化的情境	118
8.2	动态系统的模型	119
8.2.1	ARC 的符号	119
8.2.2	模型的确认	122
8.3	适于重用的体系结构	123
8.3.1	生存	123
8.3.2	递增的变化	124
8.3.3	松散耦合的组件	125
8.3.4	游戏很重要	126
8.4	结论	126
	参考文献	127
第 9 章	“方法是什么？”——关于域工程方面的一篇短文	130
9.1	引言	130
9.1.1	域、需求和软件设计	130

9.1.2	要解决的问题	131
9.1.3	愿望	131
9.1.4	本文的结构	131
9.1.5	一些格式上的约定	132
9.2	方法和方法论	132
9.2.1	方法	132
9.2.2	方法论	132
9.2.3	方法构成	132
9.2.4	方法的原理	133
9.2.5	讨论	134
9.3	域观点和方面	134
9.3.1	涉众与涉众的观点	135
9.3.2	域方面	138
9.3.3	讨论	146
9.4	结论	147
9.4.1	讨论	147
9.4.2	未来的工作	147
9.5	致谢	148
	参考文献	148

第二部分 程序设计技术

Section E	面向对象	151
第 10 章	面向对象程序设计和软件开发——一种重要的评价方法	153
10.1	引言	153
10.2	面向对象——其要求与限制	154
10.3	面向对象程序设计——评论	155
10.3.1	类规范	155
10.3.2	对象实例化	156
10.3.3	软件体系结构和组件概念	156
10.3.4	继承和多态	157
10.3.5	顺序性	157
10.4	短评面向对象分析与设计	158
10.4.1	系统模型和语义综合	158
10.4.2	软件和系统的体系结构	158
10.4.3	并发性	158
10.4.4	UML 中的数据建模	158
10.5	结论	159
	参考文献	159

第 11 章 指针和对象的痕迹模型	161
11.1 引言: 图模型	161
11.2 痕迹模型	165
11.3 应用	169
11.4 结论	173
11.5 致谢	174
参考文献	174
附录	175
第 12 章 作为堆不变量的对象模型	177
引言	177
12.1 瞬象和对象模型	178
12.2 对象模型举例	179
12.2.1 链表	179
12.2.2 记录表	180
12.2.3 记录数组	180
12.2.4 记录的集合	180
12.2.5 映射	181
12.3 关系逻辑	181
12.4 图逻辑	183
12.4.1 转换字段和分类	183
12.4.2 限定的集合	183
12.4.3 解释具体的对象模型	183
12.4.4 组合对象模型	184
12.4.5 抽象字段	184
12.4.6 索引字段	184
12.5 文本注释	184
12.5.1 表示不变量	185
12.5.2 全局不变量	185
12.5.3 抽象字段的定义	186
12.6 讨论	187
12.6.1 相关方法	187
12.6.2 相关语言	187
12.6.3 代码与问题对象模型的比较	188
12.6.4 抽象字段与抽象对象的比较	188
12.6.5 字段作为关系	189
12.6.6 限定的集合	189
12.6.7 索引字段	190
12.6.8 外延相等	190
12.7 致谢	190

参考文献	191
第 13 章 抽象依赖	193
13.1 引言	193
13.2 数据抽象的必要性	194
13.3 抽象变量的有效性	195
13.4 符号的定义	196
13.5 举例: Readers.....	199
13.6 有关工作	204
13.7 结论	205
13.8 致谢	206
参考文献	206
Section F 类型理论	209
第 14 章 类型系统	210
14.1 计算机科学中的类型系统.....	210
14.2 类型系统的用途	211
14.2.1 错误检测	211
14.2.2 抽象	212
14.2.3 文档	212
14.2.4 语言设计	212
14.2.5 语言安全	213
14.2.6 效率	214
14.2.7 更多的应用	214
14.3 历史	215
参考文献	216
第 15 章 类型的含义是什么? ——从本质到外在语义	222
15.1 语法和类型化规则	223
15.2 本质语义	224
15.3 未类型化的语义	225
15.4 逻辑关系	226
15.5 分类	230
15.6 外在的 PER 语义	233
15.7 进一步的工作与未来方向.....	235
15.8 致谢	235
参考文献	235

第三部分 应用与自动机理论

Section G 通过自动机将理论应用于实践	237
第 16 章 利用推理、探索和抽象进行自动验证	238

16.1	计算模型	239
16.2	程序行为逻辑	240
16.3	验证技术	242
16.3.1	推理法	242
16.3.2	探索性方法	243
16.4	程序和属性的抽象	244
16.4.1	数据抽象	244
16.4.2	谓词抽象	245
16.5	验证方法	248
16.6	结论	249
	参考文献	249
第 17 章	特征工程实验	253
17.1	面向特征的规范	253
17.2	特征工程面临的难题	254
17.3	面向特征的规范技术	255
17.4	一种朴素的特征工程方法	257
17.4.1	适用范围	257
17.4.2	特征和特征合成语义	258
17.4.3	特征的语法	261
17.4.4	特征交互的语义	262
17.4.5	与呼叫有关的特征交互	262
17.4.6	与状态信号有关的特征交互	263
17.4.7	与语音有关的交互	264
17.5	方法的应用	265
17.5.1	特征上下文	265
17.5.2	无约束目标特征集	266
17.5.3	特征集分析	268
17.6	评价	270
17.7	致谢	270
	参考文献	270
Section H	程序设计电路	272
第 18 章	高级电路设计	273
18.1	引言	273
18.2	电路图	274
18.3	定时	275
18.4	触发器	276
18.5	边沿触发	277
18.6	存储器	279
18.7	合并	280