

## <<Java虚拟机规范(Java SE 7)>>

### 图书基本信息

书名：<<Java虚拟机规范(Java SE 7版)>>

13位ISBN编号：9787111445159

10位ISBN编号：7111445155

出版时间：2014-1

出版时间：机械工业出版社

作者：Tim Lindholm, Frank Yellin, Gilad Bracha, Alex Buckley

译者：周志明, 薛笛, 吴璞渊, 冶秀刚

版权说明：本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问：<http://www.tushu007.com>

## &lt;&lt;Java虚拟机规范(Java SE 7)&gt;&gt;

## 内容概要

本书整合了自1999年《Java虚拟机规范（第2版）》发布以来Java世界所出现的技术变化。另外，还修正了第2版中的许多错误，以及对目前主流Java虚拟机实现来说已经过时的内容。最后还处理了一些Java虚拟机和Java语言概念的模糊之处。

针对本书，读者有任何勘误或模糊之处，均可发邮件到jvms-comments-ww@oracle.com。

2004年发布的Java SE 5.0版为Java语言带来了翻天覆地的变化，但是对Java虚拟机设计的影响则相对较小。在Java SE 7这个版本中，我们扩充了class文件格式以便支持新的Java语言特性，譬如泛型和变长参数方法等。

2006年发布的Java SE 6.0版看起来并没有为Java语言带来什么新的变化，但是对Java虚拟机的影响就比较大。

如新的字节码验证方式，它源于Eva Rose的一篇硕士论文，文中以Java Card平台为背景，展示了Java虚拟机字节码验证的另一种全新的实现思路。

这促进了Java ME CLDC第1版实现的诞生，并最终成为Java SE平台class验证过程的理论基础。关于这部分内容将会在第4章中介绍。

Sheng Liang实现了Java ME CLDC的验证器。

Gilad Bracha负责对该验证器做出详细说明，Antero Taivalsaari则是整个Java ME CLDC规范的负责人。Alessandro Coglio在字节码验证的分析方面的工作对本规范做出了很大的贡献。

Wei Tao、Frank Yellin、Tim Lindholm与Gilad Bracha一起实现的Prolog验证器是Java ME和Java SE平台规范的基础。

Wei Tao后续继续实现了实际运用于Hot Spot Java虚拟机的验证器。

之后Mingyao Yang改进了规范和设计，形成了Java SE 6中的最终实现版本。

该规范成文得益于以下JSR 202专家组成员：Peter Burka、Alessandro Coglio、Sanghoon Jin、Christian Kemper、Larry Rau、EvaRose以及 Mark Stolz。

在2011年发布的Java SE 7平台终于兑现了在1997年《Java虚拟机规范》第1版中就已做出的承诺：“在未来，我们会对Java虚拟机进行适当扩展，以便更好地支持其他语言运行于JVM之上。”

Gilad Bracha的工作是开发Java虚拟机中的热替换（hotswapping）功能，以及在Java虚拟机静态类型系统上支持动态类型语言实现。

invokedynamic指令以及支持这个指令的基础架构由John Rose以及JSR 292专家组成员：Ola Bini、Rémi Forax、Dan Heidinga、Fredrik ?hrstr?m、JochenTheodorou进行开发。

还有Charlie Nutter和Christian Thalinger做出了特别贡献。

还有许多人的名字应当出现在这里，他们在不同时间段对Java虚拟机的设计和实现做出过贡献。

我们今天所见的Java虚拟机拥有卓越的执行性能，这离不开David Ungar和他的同事们在Sun实验室Self项目中所积累的技术基础。

这些技术最初用于Self语言，后来形成了Animorphic Smalltalk虚拟机，经过长期而曲折的发展，最终成为今天Oracle HotSpot JVM的技术基础。

Lars Bak和Urs H?lzle经历了所有上述的技术发展阶段，对于今天的Java虚拟机能够拥有大家认为理所当然的高效执行性能，他们实在是居功至伟。

本规范中很多意义深远的改进来自于Martin Buchholz、Brian Goetz、Paul Hohensee、David Holmes

## <<Java虚拟机规范(Java SE 7)>>

、Karen Kinnear、Keith McGuigan、Jeff Nisewanger、Mark Reinhold、Naoto Sato、Bill Pugh、Uday Dhanikonda、Janet Koenig、Adam Messinger、John Pampuch、Georges Saab和Bernard Traversat所作出的贡献。

Jon Courtney和Roger Riggs帮助我们保证此规范的内容可同时适用于Java ME和Java SE平台。

Leonid Arbousov、Stanislav Avzan、Yuri Gaevsky、Ilya Mukhin、Sergey Reznick和Kirill Shirokov在Java技术兼容包（Java Compatibility Kit，JCK）上作出了卓越贡献，以保证本规范中描述的内容是可测试并且已测试的。

Gilad Bracha

Los Altos, California

Alex Buckley

Santa Clara, California

第2版前言

《Java虚拟机规范（第2版）》将规范所描述内容的技术背景升级到了Java 2平台（JDK 1.2），它还包括了许多对第1版的修正，并且在不改变规范内容逻辑的情况下，使描述变得更加清晰。

我们也尝试调整了规范中的字体样式、勘误（希望勘误不会产生新的错误）以及对规范中模糊的部分增加额外的描述。

另外，我们还修正了许多《Java虚拟机规范》和《Java语言规范》之间不一致的内容。

我们很感谢所有为我们梳理过第一版规范并指出问题的读者，特别感谢以下个人和团体，他们指出了问题甚至直接提供了修改意见。

Carla Schroer与她在加利福尼亚州古本蒂诺、俄罗斯新西伯利亚的兼容性测试团队（尤其感谢其中的Leonid Arbousov和Alexei Kaigorodov）。

他们煞费苦心地为第1版中各处可测试的场景编写了兼容性测试用例。

在这个过程中，他们还发现了许多处第1版规范中不清晰和不完整的内容。

Jeroen Vermeulen、Janice Shepherd、Peter Bertelsen、Roly Perera、Joe Darcy与Sandra Loosemore提交了许多有用的建议和反馈，这些建议和反馈对于第2版规范的改进工作有很大帮助。

Addison Wesley Longman出版社的编辑Marilyn Rash和Hilary Selby Polk帮助我们在第2版中合并技术变更的同时，改进了规范的可读性和内容的布局排版。

还要特别感谢Gilad Bracha，他对本书出版进行了严格审查，另外他也是本书新增内容的主要贡献者，尤其是第4、5章。

他对计算机理论的贡献以及他解决的《Java虚拟机规范》和《Java语言规范》之间的描述差异问题都极大地完善了本书。

Tim Lindholm

Palo Alto, California

Frank Yellin

Redwood City, California

1999年4月

第1版前言

《Java虚拟机规范》是描述Java虚拟机设计原理的一份完整的规范文档。

这份文档对于任何一个希望实现Java虚拟机的编译器作者，或者希望实现一个与规范兼容的Java虚拟机的程序员来说都是必不可少的。

Java虚拟机是一个抽象化的机器，整个规范中提及的Java虚拟机都是抽象化的概念，而不是特指Oracle

## <<Java虚拟机规范(Java SE 7)>>

或者其他某一家公司的Java虚拟机实现。

本书与一个具体的虚拟机实现之间的关系就犹如一份建筑蓝图与一间具体的房屋之间的关系一样。Java虚拟机具体实现（包括任何公司的JVM实现）必须包括本规范所描述的内容，但是除了少数绝对必要的地方外，本规范中的描述不应成为Java虚拟机具体实现的束缚。

我们希望这个规范至少能作为一个“实验室”版本的虚拟机实现的完整描述。

Java虚拟机源于由James Gosling在1992年设计，用于支持Oak程序语言的虚拟机。

在Java虚拟机的发展历程中，Sun公司的Green项目、FirstPerson公司、LiveOak项目、Java产品组、JavaSoft公司以及今天的Oracle公司的Java平台组中许多人都作出了直接或间接的贡献。

这本书最初是源于由Kathy Walrath编著的一份公司内部文档。

Mary Campione将本书转换为HTML版本，使大家可通过互联网访问到本书。

《Java虚拟机规范》的诞生离不开Java产品团队的总经理Ruth Hennigar的大力支持，还有编辑Lisa Friendly、Mike Hendrickson以及他在Addison-Wesley出版社的团队所做出的编辑工作。

在此特别感谢Richard Tuck对原稿的仔细审查，还有Bill Joy对本书的审查、评价和指导意义，这些宝贵信息对本书定稿都有很大的帮助。

Tim Lindholm

Palo Alto, California

Frank Yellin

Redwood City, California

1996年6月

## <<Java虚拟机规范(Java SE 7)>>

### 作者简介

从1999年4月出版《Java虚拟机规范（第2版）》至今，已经14年过去了，虽然此规范在JDK5发布的时候做了较大的更新，但却始终没有发布完整的规范。

2011年6月28日，最新的《Java虚拟机规范》终于完成并在7月份正式发布。

对于想了解Java虚拟机的程序员来说，本书是必须阅读的；想深入了解Java语言细节的程序员阅读本书也有极大好处，但是《Java虚拟机规范》、《Java语言规范》发布十余年，一直没有中文译本，这令国内不少对Java虚拟机感兴趣但英语能力较弱的程序员只能望书兴叹。

2011年年初，本书还处于草稿状态时，我就开始关注本书，并陆续对其中第1、2、6、7章进行了翻译，到2011年9月时完成了200余页的译稿。

这时候又在国内著名Java社区ItEye中结识了另外三名译者薛笛、吴璞渊和冶秀刚，我们在随后的两个多月的时间里共同完成了其余章节的翻译和校对。

在2013年，机械工业出版社获得了这本书的版权，并且采用了我们翻译的版本出版，因此有了这本中文版图书。

本书并非某一款虚拟机实现的说明书，它是一份保证各个公司的Java虚拟机实现具备统一外部接口的契约文档，书中的概念和细节描述曾经与Sun公司早期虚拟机的实现高度吻合，但是随着技术的发展，高性能虚拟机真正的细节实现方式已经渐渐与虚拟机规范所描述的内容产生了越来越大的差距。

作者也在书中不同地方反复强调：虚拟机规范中所提及的“Java虚拟机”皆为虚拟机的概念模型而非具体实现。

实现只要保证与概念模型最终等效即可，而具体实现的方式无需受概念模型束缚。

因此，通过《Java虚拟机规范》分析程序的执行语义问题（虚拟机会做什么）是十分合适且具权威性的，但分析程序的执行行为问题（虚拟机是怎样做的、性能如何）则意义不大，如果需要对具体虚拟机实现进行调优、性能分析等，我推荐在本书基础上继续阅读《Java Performance》和《Oracle JRockit：The Definitive Guide》等书。

在翻译过程中，我们尽最大努力保证作品的准确性和可读性，力求在保证语义准确的前提下，尽可能使用通俗易懂的方式向给各位读者介绍Java虚拟机的约束与运作原理。

为实现此目标，我们在翻译专有技术名词、偏僻词时用括号保留了原文，并专门在多处读者理解起来可能有困难的地方，添加了“译者注”加以解释。

囿于我们的水平和翻译时间，书中难免存在不妥之处，大家如有任何意见或建议都欢迎通过以下电子邮箱与我联系：understandingjvm@gmail.com。

周志明

## &lt;&lt;Java虚拟机规范(Java SE 7)&gt;&gt;

## 书籍目录

- 《java虚拟机规范(java se 7版)》
- 译者序
- 前言
- 第2版前言
- 第1版前言
- 第1章 引言1
  - 1.1 简史1
  - 1.2 java虚拟机2
  - 1.3 各章节摘要2
  - 1.4 说明3
- 第2章 java虚拟机结构4
  - 2.1 class文件格式4
  - 2.2 数据类型5
  - 2.3 原始类型与值5
    - 2.3.1 整数类型与整型值6
    - 2.3.2 浮点类型、取值集合及浮点值6
    - 2.3.3 returnaddress类型和值8
    - 2.3.4 boolean类型8
  - 2.4 引用类型与值9
  - 2.5 运行时数据区9
    - 2.5.1 pc寄存器9
    - 2.5.2 java虚拟机栈9
    - 2.5.3 java堆10
    - 2.5.4 方法区11
    - 2.5.5 运行时常量池11
    - 2.5.6 本地方法栈11
  - 2.6 栈帧12
    - 2.6.1 局部变量表13
    - 2.6.2 操作数栈13
    - 2.6.3 动态链接14
    - 2.6.4 方法正常调用完成14
    - 2.6.5 方法异常调用完成15
  - 2.7 对象的表示15
  - 2.8 浮点算法15
    - 2.8.1 java虚拟机和ieee 754中的浮点算法15
    - 2.8.2 浮点模式16
    - 2.8.3 数值集合转换16
  - 2.9 特殊方法17
  - 2.10 异常18
  - 2.11 字节码指令集简介20
    - 2.11.1 数据类型与java虚拟机21
    - 2.11.2 加载和存储指令23
    - 2.11.3 算术指令23
    - 2.11.4 类型转换指令24
    - 2.11.5 对象创建与操作26

## &lt;&lt;Java虚拟机规范(Java SE 7)&gt;&gt;

- 2.11.6 操作数栈管理指令26
- 2.11.7 控制转移指令27
- 2.11.8 方法调用和返回指令27
- 2.11.9 抛出异常28
- 2.11.10 同步28
- 2.12 类库28
- 2.13 公有设计, 私有实现29
- 第3章 java虚拟机编译器30
  - 3.1 示例的格式说明30
  - 3.2 常量、局部变量和控制结构的使用31
  - 3.3 算术运算35
  - 3.4 访问运行时常量池35
  - 3.5 更多控制结构示例36
  - 3.6 接收参数39
  - 3.7 方法调用39
  - 3.8 使用类实例42
  - 3.9 数组43
  - 3.10 编译switch语句45
  - 3.11 使用操作数栈46
  - 3.12 抛出异常和处理异常47
  - 3.13 编译finally语句块50
  - 3.14 同步53
  - 3.15 注解54
- 第4章 class文件格式55
  - 4.1 classfile结构56
  - 4.2 各种内部表示名称60
    - 4.2.1 类和接口的二进制名称60
    - 4.2.2 非全限定名60
  - 4.3 描述符和签名61
    - 4.3.1 语法符号61
    - 4.3.2 字段描述符61
    - 4.3.3 方法描述符63
    - 4.3.4 签名63
  - 4.4 常量池66
    - 4.4.1 constant\_class\_info结构67
    - 4.4.2 constant\_fieldref\_info、constant\_methodref\_info和constant\_interface\_methodref\_info结构67
    - 4.4.3 constant\_string\_info结构69
    - 4.4.4 constant\_integer\_info和constant\_float\_info结构69
    - 4.4.5 constant\_long\_info和constant\_double\_info结构70
    - 4.4.6 constant\_nameandtype\_info结构71
    - 4.4.7 constant\_utf8\_info结构72
    - 4.4.8 constant\_methodhandle\_info结构74
    - 4.4.9 constant\_methodtype\_info结构74
    - 4.4.10 constant\_invokedynamic\_info结构75
  - 4.5 字段75
  - 4.6 方法77
  - 4.7 属性80

## &lt;&lt;Java虚拟机规范(Java SE 7)&gt;&gt;

- 4.7.1 自定义和命名新的属性81
- 4.7.2 constantvalue属性81
- 4.7.3 code属性82
- 4.7.4 stackmappable属性85
- 4.7.5 exceptions属性91
- 4.7.6 innerclasses属性92
- 4.7.7 enclosingmethod属性94
- 4.7.8 synthetic属性94
- 4.7.9 signature属性95
- 4.7.10 sourcefile属性96
- 4.7.11 sourcedebugextension属性96
- 4.7.12 linenovertable属性97
- 4.7.13 localvariabletable属性98
- 4.7.14 localvariabletypetable属性99
- 4.7.15 deprecated属性101
- 4.7.16 runtimevisibleannotations属性101
- 4.7.17 runtimeinvisible annotations属性105
- 4.7.18 runtimevisibleparameter annotations属性106
- 4.7.19 runtimeinvisibleparameter annotations属性107
- 4.7.20 annotationdefault属性108
- 4.7.21 bootstrapmethods属性108
- 4.8 格式检查110
- 4.9 java虚拟机代码约束110
  - 4.9.1 静态约束110
  - 4.9.2 结构化约束113
- 4.10 class文件校验115
  - 4.10.1 类型检查验证116
  - 4.10.2 类型推导验证178
- 4.11 java虚拟机限制184
- 第5章 加载、链接与初始化186
  - 5.1 运行时常量池186
  - 5.2 虚拟机启动188
  - 5.3 创建和加载188
    - 5.3.1 使用引导类加载器来加载类型190
    - 5.3.2 使用用户自定义类加载器来加载类型190
    - 5.3.3 创建数组类191
    - 5.3.4 加载限制191
    - 5.3.5 从class文件表示得到类192
  - 5.4 链接193
    - 5.4.1 验证194
    - 5.4.2 准备194
    - 5.4.3 解析195
    - 5.4.4 访问控制201
    - 5.4.5 方法覆盖201
  - 5.5 初始化202
  - 5.6 绑定本地方法实现203
  - 5.7 java虚拟机退出203



## <<Java虚拟机规范(Java SE 7)>>

第6章 java虚拟机指令集204

6.1 设定：“必须”的含义204

6.2 保留操作码204

6.3 虚拟机错误205

6.4 指令描述格式205

6.5 指令集描述207

第7章 操作码助记符293

附录a limited license grant300

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:<http://www.tushu007.com>