# <<松本行弘的程序世界>>

## 图书基本信息

书名：<<松本行弘的程序世界>>

13位ISBN编号：9787115255075

10位ISBN编号：7115255075

出版时间：2011-8

出版时间：人民邮电出版社

作者：松本行弘

页数：389

译者：柳德燕,李黎明,夏倩,张文旭

# <<松本行弘的程序世界>>

## 内容概要

本书是探索程序设计思想和方法的经典之作。
作者从全局的角度，利用大量的程序示例及图表，深刻阐述了Ruby编程语言的设计理念，并以独特的视角对与编程相关的各种技术进行了考察。
阅读本书不仅可以深入了解编程世界各个要素之间的关系，而且能够学到大师级的程序思考方法。

本书面向各层次程序设计人员和编程爱好者，也可以供相关技术人员参考。

## 作者简介

松本行弘
Ruby语言的发明人，在1993年发布了Ruby语言的第一个版本，以后一直从事Ruby的设计与开发。
2011年加入著名SaaS厂商Salesforce旗下PaaS公司Heroku，任首席Ruby架构师，致力于加快Ruby Core的开发。
他还是NaCI及乐天技术研究所的研究员。
著有Ruby in a Nutshell和The Ruby Programming Language
等书。
他的博客地址为http://www.rubyist.net/~matz/。

# <<松本行弘的程序世界>>

## 书籍目录

## 章节摘录

版权页：插图：最后，变量和算式分别有自己的类型，这使得我们能够在一开始就认真考虑这些变量应该扮演什么样的角色。
我们在编写程序时就要考虑数据类型，虽然要考虑的东西变多了，但是也不能简单地说这是坏事。
显而易见，这是我们开发好的、可靠性高的程序所必需的。
从以上几点来看，静态类型似乎全都是优点。
其实它也有几个缺点，或者说是问题。
其中一个问题是，若不指定类型就写不了程序。
当然，指定类型是静态类型编程语言的特征之一。
但是说到底，数据类型只是一些辅助信息，并不是程序本质。
当我们想把精力集中到程序处理的实际问题时，却要一个个考虑数据类型的定义，这是很烦琐的。
并且，有时会让人觉得，有的类型声明仅仅是为了满足编译器的要求。
程序规模也因为数据类型的定义而变大，重要的部分反而容易被忽视。
另外一个是灵活性的问题。
静态类型本身限制了给某个变量只能赋值某种类型的对象，这种限制可能成为妨碍将来变化的枷锁。
前面学过的多重继承和接口会产生令人费解的继承关系，这时怎样设定适当的类型就变得比较困难了。
总结一下，用静态类型编程语言的人通过定义类型，把更多的信息传达了出来，这算是给编译器和将来读程序的人减轻负担的一种方法吧。
2.5.6动态类型的优点前面介绍了静态类型，那么动态类型又怎样呢？
动态类型编程语言的最大优点是源代码变得很简洁。
编程语言的进化使我们可以用更简单的程序来传达给计算机更多信息。
如果不用指定与程序本质无关的数据类型，程序也完全可以正确执行，也可以检测出来错误的话，这不是一种很好的想法吗？
得益于简洁，我们在编写程序的时候就不用考虑数据类型这些无关本质的部分了，而是可以集中于程序处理的本质部分，编写简洁程序的话，也可以提高生产力。
另一方面，有人会担心，简洁的程序虽然让我们在编程的时候变得简单了，但是因为没有类型信息，以后读起来是不是就变得难以理解呢？
虽然写的时候容易了，但难读的程序也是不可取的。
对于这样的担心，我的回答是，简洁的程序更突出了程序处理的实质，理解起来反而变得简单了。
实际上，动态类型的编程语言（例如Ruby）的程序规模和静态类型相比，程序行数相差数倍的情况并不少见。
很多人都感觉到动态类型的程序更好理解。
对于简洁程序的另外一个担心是，动态类型语言是否运行缓慢呢？
事实上是这样的。
同样的处理，在大多数情况下，静态类型编程语言运行得要快些。

# <<松本行弘的程序世界>>

## 编辑推荐

《松本行弘的程序世界》为“Ruby之父”经典力作，展现了大师级的程序思考方式。
作者凭借对编程本质的深刻认识和对各种技术优缺点的掌握，阐述了Ruby的设计理念，并由此延伸，带领读者了解编程的本质，一窥程序设计的奥秘。
《松本行弘的程序世界》不是为了介绍某种特定的技术，而是从宏观的角度讨论与编程相关的各种技术。
书中第1章介绍了作者对编程问题的新思考和新看法，剩下的内容出自《日经Linux》杂志于2005年5月到2009年4月连载的“松本编程模式讲坛”，其中真正涉及“模式”的内容并不多，大量篇幅都用于介绍技术内幕和背景分析等内容，使读者真正了解相关技术的立足点。
另外，书中还包含许多以Ruby、Lisp、Smalltalk、ErIang、JavaScrlpl等动态语言所写成的范例。
Ruby之父佳作，进入不同凡响的程序世界，深入剖析程序设计的道与术，举一反三，触类旁通。

## 版权说明