

# Python源码剖析



[Python源码剖析 下载链接1](#)

著者:陈儒

出版者:电子工业出版社

出版时间:2008-6

装帧:平装

isbn:9787121068744

作为主流的动态语言，Python不仅简单易学、移植性好，而且拥有强大丰富的库的支持。此外，Python强大的可扩展性，让开发人员既可以非常容易地利用C/C++编写Python的扩展模块，还能将Python嵌入到C/C++程序中，为自己的系统添加动态扩展和动态编程的能力。

为了更好地利用Python语言，无论是使用Python语言本身，还是将Python与C/C++交互使用，深刻理解Python的运行原理都是非常重要的。本书以CPython为研究对象，在C代码一级，深入细致地剖析了Python的实现。书中不仅包括了对大量Python内置对象的剖析，更将大量的篇幅用于对Python虚拟机及Python高级特性的剖析。通过此书，读者能够透彻地理解Python中的一般表达式、控制结构、异常机制、类机制、多线程

机制、模块的动态加载机制、内存管理机制等核心技术的运行原理，同时，本书所揭示的动态语言的核心技术对于理解其他动态语言，如 Javascript、Ruby等也有较大的参考价值。 ..

本书适合于Python程序员、动态语言爱好者、C程序员阅读

作者介绍:

目录: 第0章 Python源码剖析——编译Python

0.1 Python总体架构

0.2 Python源代码的组织

0.3 Windows环境下编译Python

0.4 Unix/Linux环境下编译Python

0.5 修改Python源代码

0.6 通往Python之路

0.7 一些注意事项

第1部分 Python内建对象

第1章 Python对象初探

1.1 Python内的对象

1.2 类型对象

1.3 Python对象的多态性

1.4 引用计数

1.5 Python对象的分类

第2章 Python中的整数对象

2.1 初识PyIntObject对象

2.2 PyIntObject对象的创建和维护

2.3 Hack PyIntObject

第3章 Python中的字符串对象

3.1 PyStringObject与PyString\_Type

3.2 创建PyStringObject对象

3.3 字符串对象的intern机制

3.4 字符缓冲池

3.5 PyStringObject效率相关问题

3.6 Hack PyStringObject

第4章 Python中的List对象

4.1 PyListObject对象

4.2 PyListObject对象的创建与维护

4.3 PyListObject对象缓冲池

4.4 Hack PyListObject

第5章 Python中的Dict对象

5.1 散列表概述

5.2 PyDictObject

5.3 PyDictObject的创建和维护

5.4 PyDictObject对象缓冲池

5.5 Hack PyDictObject

第6章 最简单的Python模拟——Small Python

6.1 Small Python

6.2 对象机制

6.3 解释过程

6.4 交互式环境

第2部分 Python虚拟机

第7章 Python的编译结果——Code对象与pyc文件

- 7.1 Python程序的执行过程
- 7.2 Python编译器的编译结果——PyCodeObject对象
- 7.3 Pyc文件的生成
- 7.4 Python的字节码
- 7.5 解析pyc文件
- 第8章 Python虚拟机框架
  - 8.1 Python虚拟机中的执行环境
  - 8.2 名字、作用域和名字空间
  - 8.3 Python虚拟机的运行框架
  - 8.4 Python运行时环境初探
- 第9章 Python虚拟机中的一般表达式
  - 9.1 简单内建对象的创建
  - 9.2 复杂内建对象的创建
  - 9.3 其他一般表达式
- 第10章 Python虚拟机中的控制流
  - 10.1 Python虚拟机中的if控制流
  - 10.2 Python虚拟机中的for循环控制流
  - 10.3 Python虚拟机中的while循环控制结构
  - 10.4 Python虚拟机中的异常控制流
- 第11章 Python虚拟机中的函数机制
  - 11.1 PyFunctionObject对象
  - 11.2 无参函数调用
  - 11.3 函数执行时的名字空间
  - 11.4 函数参数的实现
  - 11.5 函数中局部变量的访问
  - 11.6 嵌套函数、闭包与decorator
- 第12章 Python虚拟机中的类机制
  - 12.1 Python中的对象模型
  - 12.2 从type对象到class对象
  - 12.3 用户自定义class
  - 12.4 从class对象到instance对象
  - 12.5 访问instance对象中的属性
  - 12.6 千变万化的descriptor
- 第13章 Python运行环境初始化
  - 13.1 线程环境初始化
  - 13.2 系统module初始化
  - 13.3 激活Python虚拟机
- 第14章 Python模块的动态加载机制
  - 14.1 import前奏曲
  - 14.2 Python中import机制的黑盒探测
  - 14.3 import机制的实现
  - 14.4 Python中的import操作
  - 14.5 与module有关的名字空间问题
- 第15章 Python多线程机制
  - 15.1 GIL与线程调度
  - 15.2 初见Python Thread
  - 15.3 Python线程的创建
  - 15.4 Python线程的调度
  - 15.5 Python子线程的销毁
  - 15.6 Python线程的用户级互斥与同步
  - 15.7 高级线程库——threading
- 第16章 Python的内存管理机制
  - 16.1 内存管理架构
  - 16.2 小块空间的内存池

16.3 循环引用的垃圾收集  
16.4 Python中的垃圾收集  
· · · · · (收起)

[Python源码剖析\\_下载链接1](#)

标签

python

编程

Python

计算机

编译原理

程序语言

programming

程序设计

评论

我们出版的好书，即将重印。

-----  
挺好看的，对python很多机制从知其然到知其所以然。  
-----

源码可以不读，这本书还是值得读的。

---

可惜绝版了 只有电子版 不然买本来收藏

---

作者频繁地恶意卖萌，差评。

---

虽然有些不足，大段的剖析有些不简练，但总的来说提升了看问题的方式，还是挺过瘾的。一口气读完。

---

值得看

---

值得译为英文出口的精品技术专著。

---

用C讲对象python和深入理解计算机系统的python版本

---

鱼鱼送我的生日礼物哈，开心

---

常备的手边书,深入了解Python的好书

---

对python的理解又更深了一层，指导自己以后阅读更高版本的源码

---

很好的一本doc

-----  
如果再有一本ruby源码剖析，再有一本ruby和python的对照剖析。。。

-----  
有点意思

-----  
更像是其个人读书笔记，过于繁琐，陷于细节而未能抓住要领

-----  
浅显的部分看得很过瘾，关于 builtin 类型的 C 实现和 python 虚拟机机制的讲解比较容易懂，后面的类机制和内存管理机制看得很吃力，囫圇吞枣的看过算是留了点印象。源码配分析的讲解方式不错，不过错别字比较多… PS: 有种想自己实现一个语言的冲动。

-----  
虽然这本书很不错，但读过一次之后没有得到想像中的那么多的收获。作者重点介绍了几种数据类型在Python的内部实现，这是很不错的，但无论如何也看不出来这是在“深度探索”之后得到的“核心技术”。另外虽然此书能使我们对Python的内部实现有所了解，但在Python编码上却没什么帮助。

-----  
知其然亦须知其所以然

-----  
应当说是arena\_object释放arena占用的空间（实际包括所有的pool），从usable变成unused。

-----  
[Python源码剖析\\_下载链接1](#)

## 书评

Python 是公认的优秀语言，而很多人可能并不知道，标准python实现，同样是非常优秀的C源

码项目。它利用中立的标准C语言，洗练的实现了动态面向对象环境，使得Python语言的灵活和优雅得以充分体现。Python平台的C源码，本身就具体很高的学习价值。然而像其它成功的大型开源项目...

-----  
这本书其实不是面向python用户的，而是剖析python的运行时实现。更适合于进阶C程序员以提高C语言能力。  
不过不知道是作者功底不够，还是时间太紧，很多地方作者罗里罗嗦的写了好几页纸，比人家源代码还长，可还没有源代码里的注释讲得清楚。  
我仔细看过的有第十六章《Python...

-----  
好的方面已经有很多读者说过了，这里提一下我认为做得不好的几点，概括起来是“二多一缺”：二多：“口水话”多，本人认为对增进理解或过渡没有太大作用；typo多，需要质疑一下校对者的水平。  
一缺：缺少参考书目，经常读国外技术书籍的人应该不会对bibliography这个词感到...

-----  
对平台对编译器的熟悉的重要性已经不需要再强调了吧～  
虽然表面上跟很多职位没有直接的联系～个人觉得对android源码的解读 对python perl以及一切优秀开源的底层产品的解读标志着你在公司里是不是不可取代的～  
最近甚至觉得，对于拿着文档用一套API编程的位置，...

-----  
非常高兴看到又一本原创 Python 图书的出版。说起来，我和 Python 还算有一点缘分。在 2000 年的时候，非常偶然的一次机会接触到Python，当时网上资料非常之少，不知天高地厚的我竟冒失地接手了国内第一本引进 Python 图书的合作翻译工作，往事不堪回首。记得当时经常有人问...

-----  
互动网会员名：fullsail  
我在书店已经买到了，国内称得上“著”的书不多，这本书不但是著，而是著得不错。看完第一个部分就已经收益不浅啦：  
1、了解int、string，list，dict的内部机理，这样你可以避免一些很低效的使用模式。诸如多个字符串连接时如何做效率最好之类...

写python半年后的一天,看书名不错,随便买了看看,想不到物超所值,为自己省下了不少了解python的时间:

1: 文章很大篇幅讲了python虚拟机如何模拟类似x86机器语言的堆栈运行机制,很多地方作者很负责的配了流程图和测试代码。

2: 书中描述了Python设计者如何在对象分配等缓存...

我觉得这本书只是恰好用python讲而已,其实这是一本讲c跟编程语言实现的书,反正我看完了c功力见长: 内置数据结构各种课本标准实现,还有各种拿c搞继承跟多态。虚拟机挺好玩,各种模拟基于堆栈的方式。见到了传说中的GIL的实现。还有各种层面进行缓存,以及增加cache友好性。...

网上拍下了这本书,邮寄中,应该明天就到了。这也是我买的第一本Python书籍,作者研究Python源码的过程我也是经常关注的,一直等到出书。

大家好,新一期的荐书栏目如期跟大家见面了。

先来看看今天的主角是谁: 《[Python源码剖析——深度探索动态语言核心技术]》, 2008年出版,作者 @陈儒, 评分8.7分。

是的,你没看错,出版年份2008年。这本书基于Python2.5,到了十年后3.7版本的年代,可以说大部分内容已经过时...

值得一看,特别是对于有C和python基础的人,有醍醐灌顶之特效。

另外,书中两个demo代码地址失效,可在此处下载: [http://download.csdn.net/download/qg\\_25680485/10267612](http://download.csdn.net/download/qg_25680485/10267612) Python源码剖析——深度探索动态语言核心技术一书中的两个demo代码,其一是Small Python,其二是对p...

做任何事只要找对引路人,就可以快速进阶,觉得这本书就是最好的引路人。虽然曾经很想读python源码,但是一个人折腾起来效率太低,而且不知从哪入手,就一直停滞在那。遇到这本书,就像推动了第一张多米若骨牌,后续的事情就一环扣一环的展开。总结一下python解释器的精...

python之庞大不是一本400页的书可以描述的,书中省略了很多细节,基本做到详略得当,想深入的人去看源码不会有困难,不想深入的人能在书中找到python实现的一致



性 (type,class,object)。大概读了半个月，智商堪忧。python从大四用到研一结束，一直只是用而已（numpy大法好）。 ...

这本书是为数不多的, 值4星(及以上)的国内技术书. Python 解释器主要的部分都讲到了, 而且重要的地方附上图和实验说明. 没有大段泛泛而谈的介绍文字, 干货很足, 读了很有收获. 说可读性差, 一是语言不行(还老抖自以为幽默的包袱), 二是事无巨细, 没突出重点. 比如说讲闭包. 闭...

前后几个月断断续续的大致读完此书，第一感觉就是对Python有了更进一步的认识。尽管不能清楚的记住或者理解所有的细节，但是对Python虚拟机有了宏观的掌控，对写Python代码有很大的帮助。此书应该是一本典型的技术书籍（笔记），虽然书中不免口语化严重，但是确实存在太多的干...

语言的实现一直是我非常好奇的，尤其是相对静态语言更灵活的脚本语言。本书第一部分对各种内建对象的实现已经很精彩，对虚拟机的讲解让人联想起汇编，GC的讲解教我们更好的管理内存。第一遍读的时候很多系统方面的知识还不全，最后一部分没完全看懂，有时间再看一遍～

[Python源码剖析\\_下载链接1](#)