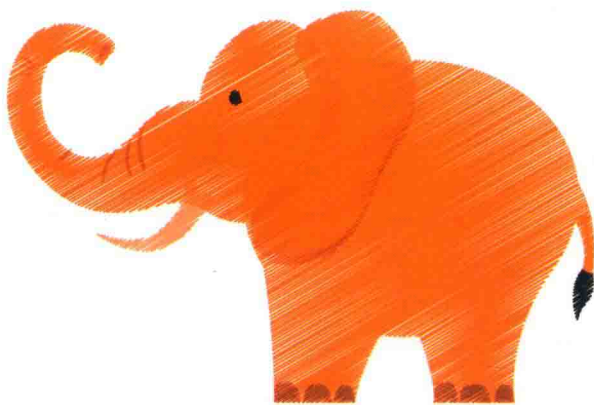




PHP 7

底层设计与源码实现

PHP 7 INTERNALS

陈雷 等编著



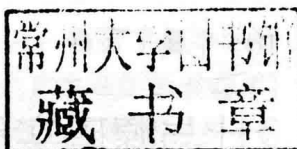
机械工业出版社
China Machine Press

PHP 7

底层设计与源码实现

PHP7 INTERNALS

陈雷 黄桃 李长林 李志 王坤 肖涛 朱栋 编著



机械工业出版社
China Machine Press

图书在版编目 (CIP) 数据

PHP 7 底层设计与源码实现 / 陈雷等编著. —北京: 机械工业出版社, 2018.6
(Web 开发技术丛书)

ISBN 978-7-111-59919-7

I. P… II. 陈… III. PHP 语言—程序设计 IV. TP312.8

中国版本图书馆 CIP 数据核字 (2018) 第 086063 号

PHP 7 底层设计与源码实现

出版发行: 机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码: 100037)

责任编辑: 高婧雅

责任校对: 殷 虹

印 刷: 北京市荣盛彩色印刷有限公司

版 次: 2018 年 6 月第 1 版第 1 次印刷

开 本: 186mm×240mm 1/16

印 张: 27.25

书 号: ISBN 978-7-111-59919-7

定 价: 99.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88379426 88361066

投稿热线: (010) 88379604

购书热线: (010) 68326294 88379649 68995259

读者信箱: hzit@hzbook.com

版权所有·侵权必究

封底无防伪标均为盗版

本书法律顾问: 北京大成律师事务所 韩光 / 邹晓东

作者简介

陈雷 滴滴顺风车运营营销研发技术负责人，曾任腾讯高级PHP研发工程师、开心网高级PHP研发工程师、百度PHP研发工程师。清华大学计算机系本科，计算机在职硕士。目前致力于滴滴顺风车智能运营平台的建设，有很强的架构能力，丰富的业务抽象经验与管理经验。

黄桃 滴滴资深开发工程师，多年PHP开发经验，尤善高并发处理、大型网站架构，乐于钻研技术难点，愿意分享。

李长林 滴滴资深工程师，曾就职于腾讯、滴滴等知名互联网公司，专注于高并发Web后台开发，活跃于开源社区。

李志 资深Web开发者，曾任职于腾讯，现就职于滴滴。

王坤 滴滴顺风车后端开发工程师，PHP爱好者，多年开发经验，喜欢学习和交流。

肖涛 穿梭在PHP各大社区多年，具有丰富的高并发处理经验，现任职滴滴高级研发工程师。

朱栋 滴滴后端开发工程师，曾多次参与大型高并发项目的设计与开发，有丰富的后端开发经验，喜欢研究新技术，喜欢折腾。

本书赞誉

编程语言是程序员表达想法的重要工具，也是很复杂的一个工具。要用好一个工具通常有几个层次：了解各种基本特性并组装完成目标、了解背后的设计原理的优点与缺点、能够改进或者创造工具。很高兴有了这本能帮助人们了解 PHP 内部实现的书籍，PHP 的代码量不小，对很多从事 Web 开发的人来说，这本书有很好的参考和学习指引作用。相信读者认真学习后会对 PHP 语言甚至各种语言有新一层次的认知，能够更好地完成自己的目标，也期待有更多的人加入改进优化 PHP 语言的行列中。

——夏绪宏，PHP 官方开发组成员、TIPI 作者

许多 PHP 程序员在熟悉 PHP 应用开发之后，开始学习 PHP 底层实现的原理，而目前关于内核实现的资料非常匮乏。本书弥补了这方面的不足。本书的特别之处是，它是一个技术团队的 PHP 内核源码学习笔记，书中内容是整个团队学习、思考、交流的结果。正在学习 PHP 内核源码的开发者可以将本书作为参考资料，结合本书内容可以更快更深入地理解 PHP 内核实现原理。

——韩天峰，车轮互联总架构师、Swoole 创始人

PHP 7 出来以后整个 PHP 的性能大幅提升，稳固了整个 PHP 语言的地位。目前市面上针对 PHP 内核分析有一些零碎的文章，但是系统分析的书籍不多。这两年除了业务的高速发展之外，滴滴公司同样也吸纳了很多 PHP 开发领域的优秀人才，最近一年也编写了 PHP 内核分析方面的书籍，确实非常值得学习和推荐。

本书对 PHP 7 内核源码分析得非常详尽，从基础代码查看方式，到源码内部数据结构扩展开发等都非常翔实，涵盖基本架构、各种数据结构、配置文件、GC、扩展开发等各类非常基础的 PHP 7 内核机制实现，也有大量代码分析，特别适合有一定 C 语言基础、爱好 PHP 内核的工程师参考学习。本书为大力推广 PHP 技术做了好榜样，值得推荐。

——谢华亮（黑夜路人）

很高兴陈雷将他和团队的学习心得与成果出版，也很荣幸第一时间阅读了一些章节，给了一些建议。我相信这是一本来自于大并发实战前沿、饱含满满诚意的作品。那些做 PHP 开发的人，如果想从“知其然”做到“知其所以然”，本书是不可不读的书籍。

——伍星，BitCV CEO

序 Foreword

PHP 7 的推出，主打的是性能提升，而且几乎可以让旧版本无缝地升级，这让很多公司直接受益，也激起了不少人探索的欲望。

作为互联网时代运用最广泛的语言之一，PHP 实战方面的书层出不穷，也不乏很多经典之作，但关于 PHP 内核分析的书少之又少。PHP 入门简单，但要看懂内部源码却需要非常深厚的功底，这也是 PHP 的特性，PHP 7 内核分析的书可以大幅度地降低分析 PHP 内部细节的门槛。

这本书最大的特点是：它是集体的智慧，是从学习过程中沉淀下来的，所以内容相对更接地气，也更简单易懂。比如详细介绍 `zval` 结构体中每个字段的含义和作用，并和 PHP 5 对比 `zval` 的变化，而且还延伸出了不少 C 的基础（如结构体和联合体、内存结构的布局），内容也很全面，从 PHP 的基本语法，到生命周期、具体的执行过程以及 PHP 的“心脏”Zend 虚拟机都有详细的分析，所以更适合纯 PHP 程序员。他们看完这本书，大体能知道一段代码经过了什么样的流程，然后被机器执行，最终输出结果，对程序员来讲，有种真正把控一切的爽快感。

这本书也有很多实用的内容，不仅有 PHP 的内部实现，还详细介绍了所有 `opcode` 的意义、`php.ini` 配置选项的作用，这些可能都可以直接应用在现有程序上，让程序变得更好。

我很欣喜地看到，不管是 PHP 官方，还是社区，都有很多的 PHP 程序员在不断突破，所以才有了 PHP 7、有了 Swoole，也有了更多关于 PHP 进阶的书，相信以后还会有更多。

最后，研读本书，相信你的 PHP 水平一定会有一个飞跃。

王晶（半桶水）

滴滴出行高级架构师

为什么要写这本书

工作了 10 多年，一直没有想过要写一本书。一年前，团队的张晶晶同学建议，我们不能组织大家每天一起学习和讨论，这样也许会成长得更快。当时我感觉这是一个非常好的建议，但是也有隐隐的担忧，因为根据经验，多个人一起学习很难长期持续。但是看到晶晶同学渴望的眼神，我决定试一试，于是我建立了一个学习圈，发给团队的同事们，他们可以自愿参加，学习的时间是每天早上 8:50~10:30，正好不影响工作，但是早晨要提前 100 分钟来公司，学习的内容是研读工作中使用的 LNMP 源码，首先学习和研究的就是 PHP 7 的源码。出乎意料，团队几乎所有的同事都参加了这个学习圈，包括实习生。

而这个学习圈一直坚持到现在，真是风雨无阻，大家已经习惯了按时一起讨论、研究，遇到问题时大家各自查找相关资料，顿悟难题时大家欣喜若狂。最令我欣喜的是，学习圈的同事们的快速成长，后来我们这个学习圈加入了滴滴公司的学习圈，得到了更多的资源支持。在学习过程中，我们每天都会记录日志，时间久了，发现得到了一个丰富的知识库，于是大家提议，写一本书，这样我们决定写一本 PHP 7 源码学习和研究的书，希望我们整理的资料能够帮助有意愿研究 PHP 7 源码的读者。

做了这个决定后，学习圈里的李长林、李志、黄桃、肖涛、王坤、季伟滨、朱栋和我一起编写了这本书，大家在工作之外，每天写书到深夜，周末从此是路人，经过半年的编写和校对，终于完成了这本书。我们希望给那些想要了解 PHP 7 底层实现的读者一些启发，帮助更多的人理解 PHP 7 的实现，也希望能够传播这种坚持学习的精神。

读者对象

- PHP 工程师
- 对 PHP 内部实现感兴趣的读者
- 有 C 语言基础的读者

□ 其他对 PHP 语言感兴趣的人

如何阅读本书

本书内容上分为四篇，共计 14 章内容。

第一篇（第 1～2 章）介绍 PHP 7 的概况和源码整体框架。其中，第 1 章主要介绍 PHP 的历史和 PHP 7 的新特性，以及研读源码的方法，第 2 章介绍 PHP 7 源码的整体框架，包括各目录的源码和功能。

第二篇（第 3～6 章）详细介绍 PHP 7 的变量实现。其中，第 3 章介绍基本变量 `zval` 的结构，第 4 章详细阐述字符串 `zend_string` 的实现，第 5 章介绍 PHP 7 中的核心结构 `HashTable` 的实现，也就是数组的实现，第 6 章介绍类和对象的实现。

第三篇（第 7～11 章）介绍 PHP 7 核心的实现。其中，第 7 章主要的内容是 PHP 7 的生命周期，第 8 章介绍生命周期的模块初始化阶段中配置文件的解析，第 9 章主要阐述内存管理的原理和实现，第 10 章介绍词法和语法分析的实现，第 11 章介绍核心 Zend 虚拟机的实现。

第四篇（第 12～14 章）主要介绍 PHP 7 中语法的实现。其中，第 12 章主要介绍基本语法的实现，第 13 章主要介绍函数的实现，第 14 章主要讲解扩展的实现。

其中，第 3 章、第 4 章、第 5 章和第 14 章为本书的重点章节，如果你没有充足的时间完成全书的阅读，可以选择性地进行重点章节的阅读。如果你是一位有着一定经验的资深人员，本书可能会是一本不错的案头书。然而，如果你是一名初学者，在开始本书阅读之前，建议先进行一些 C 语言和 PHP 的基础理论知识的学习。

勘误和支持

由于笔者的水平有限，编写时间仓促，书中难免会出现一些错误或者不准确的地方，恳请读者批评指正。如果你有更多的宝贵意见，欢迎加入我们，访问我们团队的技术网站 <https://segmentfault.com/u/php7internal> 进行专题讨论，我们会尽量在线上为读者提供满意的解答。同时，你也可以通过微博 @PHP7 内核，或者邮箱 cltf@163.com 联系到我们，期待能够得到你们的真挚反馈，在技术之路上互勉共进。

致谢

感谢卢红波、伍星两位工作导师的支持，前者是我在滴滴公司的领导，也是我管理和技术上的导师，后者是我职业生涯的启蒙导师，一直引领我前进到现在。

感谢 Swoole 团队韩天峰、王晶两位兄弟的指导和支持，他们在技术上给了我非常多的

指点。

感谢李长林、李志、黄桃、肖涛、王坤、季伟滨、朱栋 7 位兄弟在学习和研究过程中的陪伴和合作，本书是几位兄弟共同合作的结晶。

特别致谢

最后，我要特别感谢我的太太梦云、儿子和女儿，为写作这本书，我牺牲了很多陪伴他们的时间，但也正因为有了他们的付出与支持，我才能坚持写下去。

同时，感谢我的父母、岳父岳母，特别是我的父母，不遗余力地帮助我们照顾儿女，有了他们的帮助和支持，我才有时间和精力去完成写作工作。

另外要感谢我团队的兄弟们，感谢他们的坚持，为他们的成长点赞！

最后要重点感谢高婧雅编辑，得益于她的耐心审稿，宝贵的建议以及用心的修改，才进一步提升了本书的质量。

谨以此书献给我最亲爱的家人，以及众多热爱 PHP 的朋友！

陈 雷

| 内容简介 |

滴滴出行专家联合撰写，多位PHP领域大咖推荐，全面吃透PHP底层设计不二之选，全面讲解PHP内核架构、核心构成与内存管理、词法与句法解析、Zend虚拟机、函数及关键扩展等设计细节与源码实现。本书逻辑上分为四部分，共计14章内容。

第1~2章解读PHP 7的新特性、源码的整体框架（各目录的源码和功能）。

第3~6章详细介绍PHP 7的变量实现，涵盖基本变量`zval`的结构、字符串`zend_string`的实现、核心结构`HashTable`（即数组）实现，以及类与对象的实现。

第7~11章剖析PHP 7核心的实现，重点是PHP 7的生命周期，涵盖生命周期的模块初始化阶段中配置文件的解析，内存管理原理与实现，词法和语法分析的实现，核心Zend虚拟机的实现。

第12~14章介绍PHP 7中基本语法、函数、扩展的实现。

其中，第3章、第4章、第5章和第14章为本书的重点章节。

目 录 Contents

本书赞誉
序
前言

第 1 章 PHP 7 概况..... 1

- 1.1 PHP 简史与新特性..... 1
- 1.2 PHP 7 安装和调试..... 7
 - 1.2.1 编译安装 7
 - 1.2.2 使用 GDB 调试 PHP 7..... 8
 - 1.2.3 vld 扩展 10
- 1.3 PHP 7 源码阅读工具介绍..... 13
 - 1.3.1 Source Insight 13
 - 1.3.2 Understand 16
 - 1.3.3 Vim+Ctags 19
- 1.4 本章小结..... 20

第 2 章 初识 PHP 7 源码整体框架..... 21

- 2.1 PHP 7 语言的执行原理..... 21
 - 2.1.1 编译型语言与解释型语言 21
 - 2.1.2 PHP 7 的执行原理概述 23
- 2.2 PHP 7 内核架构..... 26
- 2.3 PHP 7 源码结构初步介绍..... 27

- 2.3.1 sapi 目录源码..... 27
- 2.3.2 Zend 目录源码..... 28
- 2.3.3 main 目录源码..... 29
- 2.3.4 ext 目录源码..... 29
- 2.3.5 TSRM 目录源码..... 29

2.4 本章小结..... 30

第 3 章 基本变量..... 31

- 3.1 基本知识..... 31
 - 3.1.1 数据类型 31
 - 3.1.2 结构体与联合体 32
 - 3.1.3 堆和栈的基本知识 34
- 3.2 变量的类型和实现..... 35
 - 3.2.1 PHP 5 的 zval..... 35
 - 3.2.2 PHP 7 的 zval..... 37
 - 3.2.3 PHP 7 变量类型 40
 - 3.2.4 整型和浮点型 41
 - 3.2.5 字符串类型 42
 - 3.2.6 数组 43
 - 3.2.7 引用 43
 - 3.2.8 间接 zval..... 44
 - 3.2.9 常量和常量 AST..... 46

3.2.10	资源类型	47
3.2.11	对象	47
3.3	变量的作用域	49
3.3.1	全局变量	49
3.3.2	局部变量	49
3.3.3	中间变量	50
3.3.4	静态变量	51
3.3.5	常量	51
3.4	对垃圾回收的支持	51
3.4.1	gc 的基本结构	52
3.4.2	引用计数	53
3.4.3	循环引用问题	55
3.4.4	垃圾回收	55
3.5	本章小结	63

第 4 章 字符串

4.1	字符串的结构	64
4.1.1	PHP 7 字符串的具体实现	65
4.1.2	字符串的二进制安全	68
4.1.3	zend_string API	69
4.2	智能字符串	72
4.2.1	smart_str 对比 smart_string	72
4.2.2	智能字符串的具体实现	73
4.2.3	smart_str API	75
4.3	进阶	76
4.3.1	字符串的赋值与写时分离	76
4.3.2	字符串的类别（内部字符串）	79
4.3.3	字符串的类型转换	84
4.3.4	字符串的双引号与单引号	87
4.3.5	PHP 常用字符串操作函数 实现	90

4.4	本章小结	92
-----	------------	----

第 5 章 数组的实现

5.1	基本概念	93
5.1.1	数组的语义	93
5.1.2	数组的概念	94
5.2	PHP 5 数组的实现	95
5.3	PHP 7 数组的实现	99
5.3.1	基本结构	100
5.3.2	初始化	105
5.3.3	packed array 和 hash array 的 区别	114
5.3.4	插入、更新、查找和删除	123
5.3.5	哈希冲突的解决	130
5.3.6	扩容和 rehash 操作	132
5.3.7	数组的递归保护	134
5.4	本章小结	135

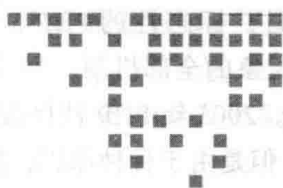
第 6 章 面向对象

6.1	类的种类	136
6.1.1	普通类	136
6.1.2	抽象类	137
6.1.3	接口	138
6.1.4	特性	140
6.1.5	final 类	142
6.1.6	匿名类	142
6.2	类的特性	143
6.2.1	类的属性	143
6.2.2	访问控制	143
6.3	类的实现	144
6.3.1	类的结构	144
6.3.2	静态属性、常量和方法	149

6.3.3 接口和特性	151	8.2 配置基本语法	203
6.3.4 继承	152	8.3 配置文件的加载	204
6.4 对象的实现	155	8.4 扩展配置	208
6.4.1 实现	155	8.5 运行时的配置	212
6.4.2 普通属性	156	8.6 本章小结	213
6.5 其他特性	157		
6.5.1 魔术方法	157	第 9 章 内存管理	214
6.5.2 自动加载	157	9.1 内存管理的意义	214
6.6 本章小结	159	9.2 内存管理的准备知识	216
		9.2.1 基本概念	216
第 7 章 生命周期	160	9.2.2 内存对齐	218
7.1 基础知识	160	9.3 内存管理的数据结构	219
7.1.1 信号处理	160	9.3.1 _zend_mm_heap	219
7.1.2 SAPI 简介	164	9.3.2 _zend_mm_chunk	223
7.1.3 SAPI 核心结构 SG(v)	166	9.4 内存管理的详细实现	226
7.2 CLI 模式的生命周期	167	9.4.1 内存管理初始化	227
7.2.1 模块初始化阶段	168	9.4.2 内存申请	227
7.2.2 请求初始化阶段	175	9.4.3 内存管理之 huge 内存	229
7.2.3 执行阶段	177	9.4.4 内存管理之 large 内存	230
7.2.4 请求关闭阶段	178	9.4.5 内存管理之 small 内存	231
7.2.5 模块关闭阶段	179	9.5 内存回收	234
7.2.6 其他工作	180	9.6 本章小结	235
7.3 FPM 模式的生命周期	181		
7.3.1 多进程管理	182	第 10 章 词法和语法分析	236
7.3.2 网络编程	193	10.1 基础知识	236
7.3.3 FastCGI 协议	195	10.1.1 编译器	237
7.4 其他模式	200	10.1.2 源程序分析	237
7.5 本章小结	201	10.1.3 BNF 范式	241
		10.2 词法与语法分析器	242
第 8 章 配置文件解析	202	10.2.1 Lex 与 YACC	242
8.1 配置文件简介	202	10.2.2 Re2c 与 Bison	247

10.3	Token 类型	252	11.3.3	设置指令 handler	307
10.4	PHP 7 词法与语法相关数据 结构	253	11.4	执行过程	308
10.4.1	CG(v) 宏	253	11.4.1	执行栈分配	309
10.4.2	zend_stack	255	11.4.2	初始化 execute_data	309
10.4.3	zend_ast 相关结构	255	11.4.3	调用 handler 函数执行	311
10.4.4	zend_arena	262	11.4.4	释放 execute_data	315
10.4.5	zend_parser_stack_elem	262	11.5	其他调度方式	315
10.5	PHP 7 词法与语法分析	263	11.6	本章小结	316
10.5.1	整体过程	263			
10.5.2	词法与语法分析阶段	263	第 12 章 基本语法实现		317
10.6	AST 的优势	273	12.1	条件判断	317
10.7	源码中的其他使用	275	12.2	循环语句	322
10.8	本章小结	275	12.2.1	foreach 语句	322
			12.2.2	while 语句	324
			12.2.3	for 语句	327
			12.2.4	do while 语句	329
第 11 章 Zend 虚拟机		276	12.3	中断与跳转	331
11.1	基础知识	276	12.3.1	break 的实现	331
11.1.1	Zend 虚拟机架构	276	12.3.2	goto 的实现	335
11.1.2	符号表	278	12.4	文件引入	338
11.1.3	函数调用栈	278	12.5	异常 / 错误处理	339
11.1.4	指令	280	12.6	本章小结	343
11.2	相关数据结构	281			
11.2.1	EG(v)	281	第 13 章 函数实现		344
11.2.2	符号表	281	13.1	基础知识	344
11.2.3	指令	284	13.2	用户定义函数的编译	345
11.2.4	指令集	288	13.3	用户定义函数的执行	352
11.2.5	执行数据	290	13.4	内置函数	358
11.2.6	执行栈	291	13.4.1	内置函数的注册	358
11.3	AST 编译过程	294	13.4.2	内置函数的执行	360
11.3.1	op_array 初始化	294	13.5	本章小结	360
11.3.2	AST 编译	295			

第 14 章 扩展	361	
14.1 扩展的实现原理.....	362	
14.2 PHP 扩展.....	363	
14.2.1 扩展的实现.....	363	
14.2.2 JSON 扩展.....	366	
14.3 Zend 扩展.....	367	
14.3.1 Zend 扩展的实现.....	367	
14.3.2 opcache 扩展.....	369	
14.4 自定义扩展.....	378	
14.4.1 初始化.....	379	
14.4.2 编译配置.....	380	
14.4.3 功能实现.....	381	
14.4.4 注册配置项.....	383	
14.4.5 编译、安装.....	385	
14.5 本章小结.....	386	
附录 A php.ini 配置项说明	387	
附录 B Token 与 opcode 含义 对应表	413	



PHP 7 概况

本章将会介绍 PHP 简史与新特性，以及 PHP7 安装和调试方法。同时还会介绍研究 PHP 7 源码的一些工具，希望对读者阅读和研究 PHP 7 源码有一定的帮助。

1.1 PHP 简史与新特性

PHP 是一种跨平台开源语言，也是迄今为止最流行的 Web 开发语言，全球有超过 80% 的网站由 PHP 驱动。PHP 自 1994 年由 Rasmus Lerdorf 创建以来已走过 20 多年，经历了 6 个大版本的更迭。下面就来了解 PHP 简史与 PHP 7 带来的新特性。

1. PHP 简史

PHP 最初是作为工具包出现的，作者 Rasmus Lerdorf 为了在自己的网站上追踪访客开发了 PHP 的雏形。而后随着功能的增多，作者发布了第一个完整的版本并称之为 Personal Home Page Tools。1996 年 Rasmus Lerdorf 发布了 2.0 版本，这是一个相对完善的版本，不仅可以访问数据库还可以把 PHP 代码嵌入到 HTML 页面中。2.0 版本吸引了很多开发者，其中包括后来 Zend 引擎的核心开发者 Zeev Suraski 和 Andi Gutmans。Zeev 和 Andi 加入之后，重写了代码，带来了 PHP 3.0。

PHP 3.0 强大的可扩展性吸引了更多的开发者加入并提交新的模块。从这个版本开始，PHP 被重命名为 PHP : Hypertext Preprocessor。2000 年夏初，4.0 版本发布，Zend 引擎正式登场，相比 PHP 3.0，最高可以有近 10 倍的性能提升。此外还支持了更多的 Web 服务器。这个时候的 PHP 已经是很流行的编程语言了，但是相对于其他语言还缺乏一些关键特性，例如面向对象、异常处理等。

2004 年 7 月，标志性的 PHP 5 正式发布，Zend 引擎升级到 2.0。PHP 5 的最大特点是引入了面向对象的全部机制。另外引进了类型提示和异常处理机制，能更有效地处理异常和避免错误。2005 年 PHP 社区发起了 PHP 6 的项目，主要的目的是为 PHP 引擎增加 Unicode 支持，但是由于种种原因，项目最终被取消。这个项目虽然被取消了，但是大量的功能陆续都添加进了 PHP 5.x 版本，例如命名空间、匿名函数、闭包等特性。

2015 年夏天，备受瞩目的 PHP 7 发布了第一个 Alpha 版本。之后，经过大概 3 个 Beta 版本和 8 个 RC 版本，2016 年 1 月 PHP 7 正式发布。PHP 7 是 PHP 一个非常重要的版本，相对于 PHP 5.x 版本，有着非常大的革新，尤其是在性能方面。如果读者的网站使用的是 PHP 5.x，那么使用 PHP 7 后几乎将无成本地得到一倍的性能提升。感谢开发者！

下面我们来测试一下 PHP 7 性能到底提升了多少。本地环境下以相同的编译参数分别安装 PHP 5.5.38、PHP 7 的第一个正式版本 7.0.2 和 7.1.0 版本，在 CLI 模式下运行 PHP 源码中的基准测试脚本。

1) 测试环境：本地搭建的 vagrant 虚拟机，操作系统 CentOS 7，单核 CPU 2.00GHz，内存 1GB。

- 2) 基准测试指标：
- ❑ Time——执行时间，以秒为单位；
 - ❑ %rel, gain——相对于上一版本节省的执行时间；
 - ❑ %abs, gain——与 PHP5.5.38 相比，脚本节省的执行时间。
- 测试结果如表 1-1 所示。

表 1-1 测试结果

PHP 版本	bench.php			micro_bench.php		
	time	%rel, gain	%abs, gain	time	%rel, gain	%abs, gain
5.5.38	2.020s	—	—	9.471s	—	—
7.0.2	0.781s	61.34%	2.59	4.193s	55.73%	2.26
7.1.0	0.688s	11.91%	2.94	3.414s	18.58%	2.77
7.1.0, 开启了 opcache	0.450s	34.59%	4.49	2.967s	13.10%	3.19

由上边的测试结果可以看出来，PHP 7.1.0 的基准性能几乎是 PHP 5.5.38 的 3 倍左右，在开启了 opcache 的情况下更是达到了 4.4 倍之多，这是一个非常显著的提升。这些性能提升是如何做到的呢？本书后续的章节将一一介绍。

 **注意** 这里的测试是纯 CPU 的基准测试，5 次运行取平均值，不包括其他方面的测试，在实际的项目或者其他运行环境下可能有所差异。

2. PHP 7 新特性

PHP 7 除了在性能方面有极大提升外，还添加了很多新的特性，如太空船操作符、标量