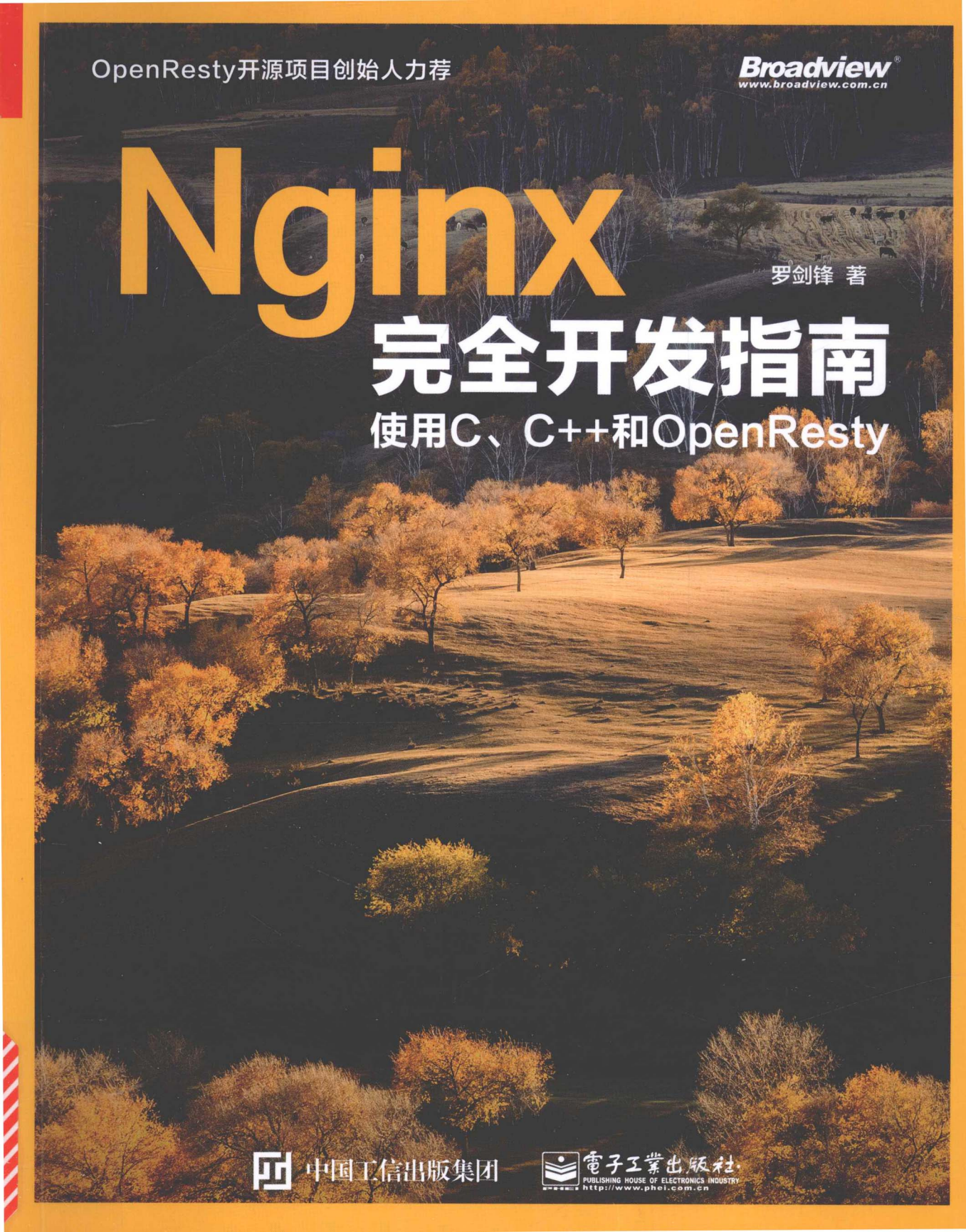
OpenResty开源项目创始人力荐
Broadview
www.broadview.com.cn
Nginx
罗剑锋 著
完全开发指南
使用C、C++和OpenResty
中国工信出版集团
電子工業出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
http://www.phei.com.cn

Nginx
完全开发指南
使用C、C++和OpenResty

罗剑锋 著

電子工業出版社
Publishing House of Electronics Industry
北京•BEIJING

内 容 简 介

Nginx 是著名的 Web 服务器，性能优异，运行效率远超传统的 Apache、Tomcat，广泛应用于国内外诸多顶级互联网公司。

Nginx 的一个突出特点是其灵活优秀的模块化架构，可以在不修改核心的前提下增加任意功能，自 2004 年发布至今，已经拥有百余个官方及非官方的功能模块（如 proxy、mysql、redis、rtmp、lua 等），使得 Nginx 成长为了一个近乎“全能”的服务器软件。

Nginx 功能强大，架构复杂，学习、维护和开发的门槛较高。为了帮助读者跨越这一障碍，本书深入最新的 Nginx 源码（Stable 1.12.0），详细剖析了模块体系、动态插件、功能框架、进程模型、事件驱动、线程池、TCP/UDP/HTTP 处理等 Nginx 核心运行机制，在此基础上讲解如何使用 C、C++、Lua、nginScript 等语言来增强扩展 Nginx，让任何人都能够便捷、轻松地开发和定制 Nginx，进而应用到自己的实际工作中，创造出更多的价值。

本书结构严谨、脉络清晰、论述精确、详略得当、图文并茂，值得广大软件开发工程师、系统运维工程师和编程爱好者拥有。

图书在版编目（CIP）数据

Nginx 完全开发指南：使用 C、C++和 OpenResty / 罗剑锋著. —北京：电子工业出版社，2017.6
ISBN 978-7-121-31457-5

Ⅰ. ①N… Ⅱ. ①罗… Ⅲ. ①互联网络－网络服务器－程序设计－指南②C 语言－程序设计－指南
Ⅳ.①TP368.5-62②TP312.8-62

中国版本图书馆 CIP 数据核字(2017)第 094580 号

策划编辑：孙学瑛
责任编辑：安　娜
印　　刷：北京京科印刷有限公司
装　　订：北京京科印刷有限公司
出版发行：电子工业出版社
　　北京市海淀区万寿路 173 信箱　邮编：100036
开　　本：787×980　1/16　印张：39.75　字数：901 千字
版　　次：2017 年 6 月第 1 版
印　　次：2017 年 6 月第 1 次印刷
定　　价：99.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：（010）88254888，88258888。

质量投诉请发邮件至 zlts@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式：010-51260888-819，faq@phei.com.cn。

前言

缘起

最早接触 Nginx 大概是在 2011 年，面对着一个全新的 Web 服务器，和大多数人一样最初我也是一片茫然，能找到的参考资料十分有限，安装、配置、运行几乎都是“摸着石头过河”，犯过许多低级错误。

随着对 Nginx 逐渐熟悉，它的高并发处理能力给我留下了深刻的印象，作为一个开源软件的爱好者，很自然地想要探究一下它的内部工作原理。我由此开始了对 Nginx 源码的钻研之路，中间经过了很多的艰辛曲折，走过不少的弯路。

我最常用的工作语言是 C++，所以在阅读 Nginx 源码时也总以 C++的面向对象方式来思考和理解，以对象作为切入点记笔记、画 UML：从最简单的 ngx_str_t、ngx_array_t 入手，然后到 ngx_request_t、ngx_upstream_t 等复杂的结构，再围绕着这些对象研究相关的功能函数和处理流程，梳理代码逻辑的同时也摸索着使用 C++编写 Nginx 模块的方法，逐渐积累了一些用起来颇为顺手的小工具——当然还是比较初级的形式。

三年多前，我被调到了新的工作岗位，需要重度使用 Nginx 开发，这让我以前的零散积累终于有了用武之地。那段时间里使用 C/C++陆续做了很多东西，也借着机会重新优化了原有的工具代码。

繁忙的工作之余，我有了种进一步整理经验的迫切感，因为只有系统完整地分享这些知识，才能让更多的人基于 Nginx 二次开发，让 Nginx 更好地为网络世界服务。

同一时间，市面上也出现了一些 Nginx 开发相关的资料、书籍，但在我看来却有“粗制滥造”之嫌：行文混乱，“车轱辘话”“口头禅”满天飞，甚至大段照抄指令说明，还有对源码

的曲解，未免有点儿“误人子弟”，读起来实在是难受。终于，在“忍无可忍”的心态之下，我动起了写作本书的念头。

经过近一年的努力，现在这本书终于呈现在了读者面前，结构上基本反映了我学习研究 Nginx 时的心路历程，从最初的“一无所知”起步，逐渐深入到定制开发的层次，希望能与读者“心有戚戚焉”。

Nginx 随感

毫无疑问，Nginx 是目前这个星球上所能获得的最强劲的 Web 服务器（没有之一），同时也是目前最成熟、最优秀的 TCP/HTTP 服务器开发框架。

Nginx 资源消耗低，并发处理性能高，配置灵活，能够连接 CGI、PHP、MySQL、Memcached 等多种后端，还有着出色的负载均衡能力，可以整合封装各种 service，构建稳定高效的服务。如今 Nginx 已经成为了网站架构里不可或缺的关键组件，广泛应用于国内外许多大型 IT 企业。每一个繁忙的网站背后，可能都有 Nginx 默默工作的身影。

在 Nginx 出现之前，使用 C/C++开发 Web 服务器是项比较“痛苦”的工作，虽然有很多网络程序库可以使用（例如 asio、libevent、thrift 等），但它们通常只关注较底层的基础功能实现，离成熟的“框架”相距甚远，不仅开发过程烦琐低效，而且程序员还必须要处理配置管理、进程间通信、协议解析等许多 Web 服务之外的其他事情，才能开发出一个较为完善的服务器程序。但即使开发出了这样的服务器，通常性能上也很难得到保证，会受到程序库和开发者水平等因素的限制——很长一段时间里，C/C++在 Web 服务器领域都没有大展拳脚的机会。

Nginx 的横空出世为 Web 服务器开辟了一个崭新的天地，它搭建了一个高性能的服务器开发框架，而且是一个完整的、全功能的服务器。模块化的架构设计很好地分离了底层支撑模块和上层逻辑模块，底层模块处理了配置、并发等服务器的外围功能，核心支撑模块定义了主体的 TCP/HTTP 处理框架。开发者可以把大部分精力集中在上层的业务功能实现上，再也不必去为其他杂事而分心，提高了软件的开发效率。

在 Nginx 框架里，C/C++程序员可以尽情发挥自己的专长，充分利用 Nginx 无阻塞处理的优势，打造出高质量的 Web 应用服务器，与其他系统一较高下。

Nginx 和 C/C++

Igor Sysoev 选择用 C 语言（准确地说是 ANSI C）来实现 Nginx 肯定是经过了认真

的考虑。

作为与 UNIX 一同诞生的编程语言，C 语言一直是系统级编程的首选。和其他高级语言相比，它简单可靠，更接近计算机底层硬件，运行效率更高。指针更是 C 语言的一大特色，善用指针能够完成许多其他语言无法完成的工作。

以 C 语言实现的 Nginx 没有“虚拟机”的成本，省略了不必要的中间环节，直接操纵计算机硬件，从根本上提高了 Web 服务器的处理能力。虽然 C 语言不直接支持面向对象，但 Nginx 灵活运用了指针，采用结构体+函数指针的形式，达到了同样的效果，从而使软件拥有了良好的结构。

C++是仅次于 C 的系统级编程语言，在兼容 C 的同时又增加了类、异常、模板等新特性，还支持面向对象、泛型、函数式、模板元等多种编程范式，可以说是计算机语言里的一个“庞然大物”。C++的特性很多，有的也很好用，但总体上的确比较复杂，易学难精，容易被误用和滥用，导致低效、难维护的代码，我想这可能是 Igor Sysoev 放弃使用 C++的一个重要原因。

另一个可能的原因是 C 语言本身已经非常稳定，几十年来没有太大的变动，在各个系统里都支持得非常好。而 C++在 1998 年才有了第一个标准，并且现在还在发展之中，语言特性还不够稳定（例如 export、register 等曾经的关键字在 C++11 里就已经被废弃），许多编译器对 C++的支持程度也有差异，这与 Nginx 的高可移植性目标明显不符。

但 C++毕竟还是有很多的优点，类可以更好地封装信息、异常简化了错误处理、模板能够在编译期执行类型计算。在 C++11 标准颁布之后，C++更是几乎变成了一门“全新”的语言，auto/decltype/nullptr/noexcept 等新关键字增强了语言的描述能力，标准库也扩充了相当多的组件，易用性和稳定性都大大提升。

在 Nginx 里使用 C++时要对 C++的长处和不足有清醒的认识，避免多层次继承、虚函数等影响效率的编程范式，只使用经过充分验证的、能够切实提高开发效率和性能的语言特性和库，避免华而不实的技术炫耀，尽量做到像 Nginx 源码那样质朴踏实。只有这样，才能够发挥出 1+1>2 的作用，让 Nginx 从 C++中得到更进一步的发展动力。

Nginx 和 OpenResty

多年以前 Nginx 开发使用的语言只能是 C 和 C++，而现在，越来越多的开发者逐渐转向了 OpenResty，使 Lua 搭建高并发、高性能、高扩展性的 Web Server。

我接触 OpenResty 的时间并不算很长，大约在四年左右。由于 C/C++程序员“天生的傲

慢”，一开始对 OpenResty 确实有点儿“抵触情绪”，总觉得脚本程序比不上 C/C++实现。然而随着使用的增多，特别是在研究了它的源码之后，我不得不感慨 OpenResty 的精致、完美和强大，简直是所有 Nginx 开发者“梦寐以求的至宝”。

由于 agentzh 对 Nginx 的运行机制了如指掌，OpenResty 的核心部分——ngx_lua 一个模块就涵盖了 access/rewrite/content/log 等多个处理阶段，再搭配上小巧灵活的 Lua 和高效的 LuaJIT，我们就能够在更高级的业务层次上使用“胶水”代码来调用组合 Nginx 底层功能，轻松开发出丰富 Web 服务，极大地节约了宝贵的时间和精力。

当然，OpenResty 并不只有 ngx_lua，围绕着 ngx_lua 还有众多的库和辅助工具，构成了一个相当完善的生态环境，这些组件相互支撑，利用得当可以更好地提高生产效率。

OpenResty 现在正处于蓬勃发展的阶段，今后的 OpenResty 也许不仅限于 Nginx 和 Web Server，而将成为一个更通用的开发平台，工作语言也不仅限于 Lua，可能还会有其他新的语言（例如 agentzh 正在做的 edgelang 和 fanlang），让我们拭目以待。

致谢

首先当然要感谢 Nginx 的作者 Igor Sysoev，没有他就不会有如此优秀的 Web 服务器，也就不会有本书的诞生。

OpenResty 创始人章亦春（agentzh）是一位非常亲切随和的人，在 Nginx、DSL、Dynamic Tracing 等领域造诣极高，本书部分章节有幸经他审阅，在此表示最诚挚的谢意。

亲情永远是人生命中最值得珍惜的部分，我要感谢父母多年来的养育之恩和“后勤”工作，感谢妻子在生活中的陪伴，感谢两个可爱的女儿，愿你们能够永远幸福快乐。

最后，我也要感谢读者选择本书，希望读者能够在阅读过程中有所收获，在 Nginx 开发过程中获得乐趣。

您的朋友 罗剑锋

2017 年 4 月 28 日 于 北京 亚运村

目录