

O'REILLY®



Google App Engine 编程 (Java版)

Programming Google App Engine with Java

中国电力出版社

Dan Sanderson 著
方红琴 冯花平 译

Google App Engine编程(Java版)

Dan Sanderson 著

方红琴 冯花平 译

Beijing • Cambridge • Farnham • Köln • Sebastopol • Tokyo

O'REILLY®

O'Reilly Media, Inc. 授权中国电力出版社出版

中国电力出版社

图书在版编目 (CIP) 数据

Google App Engine编程: Java版 / (美) 丹·桑德森 (Dan Sanderson) 著; 方红琴, 冯花平译. -- 北京: 中国电力出版社, 2016.10

书名原文: Programming Google App Engine with Java

ISBN 978-7-5123-9400-1

I. ①G… II. ①丹… ②方… ③冯… III. ①网页制作工具—JAVA语言—程序设计
IV. ①TP393.092

中国版本图书馆CIP数据核字 (2016) 第115979号

北京市版权局著作权合同登记

图字: 01-2016-2779号

Copyright © 2015 Dan Sanderson, All right reserved.

Simplified Chinese Edition, jointly published by O'Reilly Media, Inc. and China Electric Power Press, 2016.
Authorized translation of the English edition, 2015 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由O'Reilly Media, Inc. 出版2015。

简体中文版由中国电力出版社出版2016。英文原版的翻译得到O'Reilly Media, Inc.的授权。此简体中文版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc.的许可。

版权所有, 未得书面许可, 本书的任何部分和全部不得以任何形式复制。

封面设计/ Ellie Volckhausen, 张健

出版发行/ 中国电力出版社 (<http://www.cepp.sgcc.com.cn>)

地 址/ 北京市东城区北京站西街19号 (邮政编码100005)

经 销/ 全国新华书店

印 刷/ 北京天宇星印刷厂

开 本/ 787毫米×980毫米 16开本 23.75印张 449千字

版 次/ 2016年10月第一版 2016年10月第一次印刷

印 数/ 0001—3000册

定 价/ 69.00元 (册)

敬告读者

本书封底贴有防伪标签, 刮开涂层可查询真伪
本书如有印装质量问题, 我社发行部负责退换

版 权 专 有 翻 印 必 究

O'Reilly Media, Inc.介绍

O'Reilly Media通过图书、杂志、在线服务、调查研究和会议等方式传播创新知识。自1978年开始，O'Reilly一直都是前沿发展的见证者和推动者。超级极客们正在开创着未来，而我们关注真正重要的技术趋势——通过放大那些“细微的信号”来刺激社会对新科技的应用。作为技术社区中活跃的参与者，O'Reilly的发展充满了对创新的倡导、创造和发扬光大。

O'Reilly为软件开发人员带来革命性的“动物书”；创建第一个商业网站（GNN）；组织了影响深远的开放源代码峰会，以至于开源软件运动以此命名；创立了Make杂志，从而成为DIY革命的主要先锋；公司一如既往地通过多种形式缔结信息与人的纽带。O'Reilly的会议和峰会集聚了众多超级极客和高瞻远瞩的商业领袖，共同描绘出开创新产业的革命性思想。作为技术人士获取信息的选择，O'Reilly现在还将先锋专家的知识传递给普通的计算机用户。无论是通过书籍出版，在线服务或者面授课程，每一项O'Reilly的产品都反映了公司不可动摇的理念——信息是激发创新的力量。

业界评论

“O'Reilly Radar博客有口皆碑。”

——Wired

“O'Reilly凭借一系列（真希望当初我也想到了）非凡想法建立了数百万美元的业务。”

——Business 2.0

“O'Reilly Conference是聚集关键思想领袖的绝对典范。”

——CRN

“一本O'Reilly的书就代表一个有用、有前途、需要学习的主题。”

——Irish Times

“Tim是位特立独行的商人，他不光放眼于最长远、最广阔的视野并且切实地按照Yogi Berra的建议去做了：‘如果你在路上遇到岔路口，走小路（岔路）。’回顾过去Tim似乎每一次都选择了小路，而且有几次都是一闪即逝的机会，尽管大路也不错。”

——Linux Journal

目录

前言	1
第1章 谷歌App引擎简介	11
运行时环境	12
静态文件服务器	14
前端高速缓存	14
云数据存储	15
服务	18
谷歌账户、OpenID和OAuth	20
谷歌云端点 (Google Cloud Endpoint)	21
任务队列和时钟守护作业 (Cron Job)	21
命名空间	22
开发工具	23
云控制台	24
开始应用程序开发	24
第2章 创建应用程序	26
创建云SDK	26
开发应用程序	33
注册应用程序	54
上传应用程序	55
测试应用程序	56
实现计费功能	58

第3章 配置应用程序	59
应用程序引擎结构.....	60
配置Java应用程序.....	62
应用程序的ID和版本.....	64
多线程.....	66
请求处理器.....	66
静态文件和资源文件.....	67
域名.....	70
谷歌应用程序.....	72
配置安全连接.....	75
与定制域的安全连接.....	76
对谷歌账户认证.....	79
环境变量.....	80
入站服务.....	80
定制错误响应.....	81
Java服务器端小程序会话.....	82
第4章 请求处理器和实例	85
运行时环境.....	86
请求处理器抽象.....	93
引入实例.....	96
请求计划和等待延迟.....	98
预热请求.....	99
常驻实例.....	100
实例类与应用.....	101
实例小时和计费.....	102
实例控制台面板.....	103
流量分割.....	104
第5章 使用模块	105
实例布局.....	106
配置模块.....	107

手动扩展和基本扩展	113
启动请求	114
关闭钩子	115
后台线程	116
模块和开发服务器	117
部署模块	118
使用URL定位模块	119
将请求分派给模块	122
启动和停止模块	123
管理和删除模块和版本	124
模块API	125
始终运行的实例	126
第6章 数据存储实体	130
实体、键和属性	131
Java Datastore API简介	133
属性值	135
键和键对象	139
实体的使用	141
分配系统ID	144
开发服务器和数据存储	145
第7章 数据存储查询	146
查询和类型	147
查询结果和键	147
查询API	148
索引简介	153
自动索引和简单查询	155
定制索引和复杂查询	161
不相等和IN过滤器	167
未设置的属性和未索引的属性	169
排序顺序和值的类型	170

查询和多值属性	171
查询游标	179
投影查询	182
配置索引	185
第8章 数据存储事务.....	187
实体和实体组	189
事务的具体执行过程.....	193
Java中的事务.....	195
如何更新实体	199
如何读取实体	201
批量更新	202
如何更新索引	203
跨实体组的事务	204
第9章 数据存储管理.....	206
监测数据存储	206
管理索引	208
从应用程序访问元数据	210
远程控制	215
第10章 Java持久化API.....	219
创建JPA	220
实体和键	221
实体属性	224
嵌入对象	225
存储、获取和删除对象	226
JPA中的事务	228
查询和JPQL.....	229
关系	232
如何获取更多信息.....	237

第11章 在App Engine中使用Google Cloud SQL.....	238
选择Cloud SQL实例	239
本地安装MySQL	240
创建Cloud SQL实例	241
从自己的计算机连接至实例	242
创建数据库	244
创建JDBC	246
从App Engine连接至数据库	247
备份和恢复	254
导出和导入数据	254
gcloud sql命令	255
第12章 存储高速缓存.....	258
使用Java调用存储高速缓存	260
键和值	261
设置值	261
获取值	263
删除键	263
原子增加和减少	263
比较并设置	264
对存储高速缓存服务的批调用	265
存储高速缓存和数据存储	267
处理高速缓存错误	267
存储高速缓存管理	268
第13章 获取URL和网络资源	271
获取URL	272
出去的HTTP请求	275
SSL上的HTTP (HTTPS)	277
请求和响应长度	277
请求期限	278
处理重定向	278

响应对象	279
第14章 发送和接收Email消息	280
发送Email消息	281
接收Email消息	288
第15章 使用XMPP发送和接收即时消息	291
邀请用户聊天	293
发送聊天信息	294
接收聊天消息	296
处理错误消息	298
管理所处的状态	299
第16章 任务队列和被调度的任务	307
配置任务队列	310
使任务入队	311
任务参数	313
入队队列	316
出队队列	321
采用事务方式将任务入队	324
任务链	327
任务队列管理	332
延后执行工作	332
计划任务	333
第17章 优化服务调用	337
异步调用服务	338
使用AppStats进行可视化调用	342
第18章 管理请求日志	349
记录日志	350
查看最近的日志	352

下载日志	354
保留日志	355
从应用程序查询日志	355
第19章 部署和管理应用程序	358
上传应用程序	359
使用版本	359
管理服务配置	361
App Engine设置	362
管理开发人员	363
配额和计费	363
获取帮助	364

前言

在因特网上，今天的流行有可能一闪即逝。如果流行新闻网站上提到了你的网站，立刻会带来300000潜在的客户，这些客户都希望知道你是何许人也，并且想知道你能够提供什么。但是，如果是一个刚起步的小公司，公司的硬件和软件不可能处理如此大的网络流量。实际上，公司精心创建的网站在公司起步的前6个月能够处理每小时30000次访问。在高负载的情况下，此系统甚至无法向其余270000个想浏览网站的用户显示公司的logo。即使在访问流量减少之后，这些潜在的客户也不可能再回来访问网站了。

解决这个问题的方案并不是公司创建的第一天就花时间和成本建立能为上百万用户服务的系统，因为在接下来的几个月内，系统可能每天仅需要为数千用户提供服务。如果延迟开发大的系统，可能会错失利用顾客反馈改善产品的良机。如果早期就创建大的系统，开发的系统中可能会包含用户不需要的功能。

从历史上看，小公司不会在第一天就访问大的服务器系统。它们的最佳选择是创建一个小的系统，并且希望随着公司的增长，这个小系统的停用不会影响到公司的声誉。有些幸运的公司会找到自己的受众，获得另一轮融资，并且终止开发新的特征来创建大容量的产品。当然，有些公司并不是这么幸运。

现在，这些公司有了另外的出路。大的互联网公司（例如，亚马逊、谷歌和微软）会以按次计费的模式租赁部分大容量系统。公司的网站可以安装在这些大系统上提供服务，这些大系统足以处理突如其来的大流量，并且运行非常成功。并且因为仅按使用情况付费，所以不存在当流量低时浪费成本的超前投资。随着用户群的增长，成本也会按比例增长。

谷歌公司提供的大系统统称为谷歌云平台（Google Cloud Platform），它由一组高性能

的服务和工具构成，包括各种级别的虚拟机、多种形式的可靠数据存储、可配置的组网、自动扩展的基础设施，甚至可以包括支撑谷歌产品的大数据分析工具。但是谷歌云平台并不仅仅只提供对谷歌基础设施的访问，它封装了应用架构最好的实际应用。谷歌的工程师已经在他们自己的产品中反复地使用过这些应用架构。

谷歌云平台的中心部件是谷歌App引擎，它是托管自动增长服务的应用程序。App引擎运行应用程序，使得无论数十个类似的用户还是上千个类似的用户访问应用程序，每个用户与其他用户都能获得相同的体验。应用程序代码只需关注用户的个人体验。App引擎自动完成大规模的计算任务，诸如负载均衡、数据复制和容错处理。

在传统系统的第一个数据库服务器过度增长的情况下，这种可扩展模式得到了真正的发展。使用此系统，添加负载均衡的网络服务器和高速缓存层可以让你的应用程序实现更多的功能，但是当应用程序需要在多个位置写数据时，会面临大的难题。当应用开发到达依赖数据库软件特征的阶段，并且数据库软件无法管理跨越多台机器分布的数据时，问题尤为严重。如果事先考虑根据云平台模式布局数据，以后就不必重新创建整个系统了。

通常会被人们忽视的一个优点是，App引擎的执行模式有助于分配计算及数据。App引擎擅于将计算资源迅速分配成小的任务。这种方式最初设计用于在优先考虑响应客户情况下处理来自用户的网络请求。将此执行模式与云平台任务队列服务结合起来，可以将中等规模和大规模的计算任务分解成可并行执行的信息块。如果任务执行失败，系统会不断地重新尝试执行这些任务，直到成功为止，这使得在服务失败的情况下能够更有弹性地执行任务。这种执行模式使得人们积极优化平台的并行性和强壮性。

在谷歌基础设施上运行意味着不需要搭建服务器，不需要更换失效的硬件驱动，也不需要找哪块网卡出了问题。也不会因为ISP的一点小问题导致服务器报警而被警报器半夜叫醒。并且因为可以自动扩展，所以不需要随着流量的增长而不断地添加新硬件。

谷歌云平台和App引擎让你只关注应用程序的功能和用户体验。使用云平台，可以更早地运行大系统、得到大量的关注、保持客户并且在用户的帮助下提升产品。在不需要创建新体系结构的情况下，App随着受众的规模而增长，直到与谷歌级别成一定比例为止。同时，你的竞争对手还在自己应用程序上到处灭火，或者绞尽脑汁配置数据库。

在本书的帮助下，你将学会如何开发在谷歌云平台上运行的网络应用程序，以及如何掌握App引擎可扩展执行模式的主要技能。本书的主要部分讨论谷歌云数据存储（Google Cloud Datastore），它是一种高性能的数据服务，与过去数十年在网络开发中占主导地位的关系型数据库有所区别。应用模式和数据存储一起为我们呈现了一种新的看待网络

应用程序的方式，虽然网络应用程序如我们熟知的模式一样简单，但我们需要重新思考之前忽略的一些原理。

App引擎简史

如果读完本书，你可能想知道为什么书名为《Google App Engine编程》，而不是《谷歌云平台编程》。简要地回答就是，作为一个整体，云平台的能力对于一本书来说太宽泛。具体地讲，计算引擎、平台原始虚拟机能力不仅只服务于网络应用程序，而且可以完成各种各样的功能。

通过一些账户（至少包括我的账户），App引擎开始的时候只是实现云平台思想，并且进而演化和扩展为包括大规模和可弹性规模计算。当它2008年第一次运行时，App引擎使用可扩展数据存储、任务队列服务和公共特征服务的API，托管使用Python语言编写的网络程序，这些API位于App代码运行（例如，访问网络）于其中的“容器”之外。接着，Java“运行时环境”能够使用相同的可扩展基础设施基于Java服务器端小程序运行网络应用程序。容器化的应用代码、无模式数据存储以及面向服务的架构证明其不仅为创建可扩展网络应用程序很好的方式，而且是使可靠性为App引擎产品关键部分的好的方式（不会再报假警了）。

App引擎不断进步，从功能上来说具有几个主要的里程碑。第一个主要的里程碑是数据存储的大升级，其使用新的基于Paxos的复制算法。新算法改变API保证数据一致性的方式，使得将其作为选择进入迁移（包括自动迁移工具）进行发布。另外一个重要里程碑是由通过CPU使用计费的独立请求转换成通过实例正常运行时间计费的长期运行应用实例。使用这种升级的执行模式，App代码可以将准备工作推送到用户请求逻辑之外执行，并且利用本地存储高速缓存。

谷歌将计算引擎（Compute Engine）作为独立的产品运行，这是为了通用目的按需访问计算的方式。使用计算引擎VM，可以运行任何64位基于Linux的操作系统，并且可以将任何语言编写的执行代码编译到此操作系统，或者由此操作系统进行解释。App（无论是在App引擎上运行的或采用其他方式运行的App）可以让计算引擎启动任意数量的虚拟机、完成任务，并且在不再需要时关闭机器或采用传统配置或定制配置让它们运行。

App引擎和计算引擎采用不同的方法提供不同的功能。但是这些技术已经开始融合。在2014年年初，谷歌公布托管虚拟机（Managed VM），这是一种采用类似App引擎方式运行基于VM代码的新方式（写本书时，此特征还不是完全可用，但可以检查谷歌云平台网站的更新）。总之，可以采用你所需要数量的平台来实现目标，需要的时候可以采用弹性投资的方式，并且让平台自动地处理其他的事情。

本书是在App引擎历史的转折点完成的。原来为App引擎创建的服务正被推广至云平台，并给出REST API，使得也可以从平台之外调用它们。使用新的通用云SDK和云控制台扩展App引擎开发工具。我甚至看到已开始使用新方式来开发并部署软件，即使用基于集成Git源代码版本控制的方式。已经有很多与技术进步相关的书籍，下面仅是一个概貌，主要强调主要概念和长期持续的主题。

本书主要关注使用App引擎创建网络应用程序以及与平台（尤其与云数据存储）相关的部分。我们将讨论当前App引擎独有的服务，例如获取URL及发送电子邮件的服务。我们还会讨论使用任务队列和离线进程组织和优化应用程序的技术，以及其他从谷歌App引擎获得最多的技术。

如何使用本书

本书关注App引擎的Java运行时环境。Java运行时包括完整的Java 7 JVM，它能够运行Java编译器产生的字节码，以及由以JVM为目标的其他语言的编译器产生的字节码。App引擎是J2EE标准的服务器端小程序容器，并且包括用于Java网络开发的标准库和特征。你甚至可以将App部署为WAR文件。

App引擎支持3种其他运行时环境：Python、PHP和Go。Python环境为Python编程语言提供快速的解释器，并且与许多Python开放源网络应用程序框架兼容。PHP环境使用标准库运行原生PHP解释器并且可以将其进行许多扩展，并且能够运行许多现有的PHP应用程序，诸如WordPress 和Drupal。使用Go运行时环境，App引擎编译服务器上的Go代码，并且以CPU本来的速度执行这些代码。

原本打算在一本书中包含全部相关信息（其中也包括Python）。为了让读者更容易找到自己熟悉语言相关的信息，将此书分成具体语言的版本。本书是Java版的。《Google App Engine编程（python 版）》包括相同的内容，但其使用Python语言，并包括与Python相关的具体主题。

将来有可能写本书的PHP版本和Go版本。但是现在，官方的App引擎文档是在平台上使用这些语言最好的资源。如果你对本书的PHP版和Go版感兴趣，可以将邮件发送至bookquestions@oreilly.com。

本书的组织方式便于读者跳至自己最关心的主题进行阅读。“第2章创建应用程序”介绍了一个应用程序的布局，此实例使用了多个特征，读者可以从这个完整的实例开始学习。接下来的章节介绍了App引擎的各种特征，主要关注有效的数据存储和恢复、通信和分布式计算。此外，还涉及项目生命周期相关的主题，例如项目的发布和维护。

云数据存储是一个非常大的主题，因此使用多个章节介绍相关内容。从第6章开始，介

绍了数据存储概念及与这些概念相关的Java API。本书还使用一章介绍了可以和云数据存储一起使用的JPA、标准接口以及可能访问数据存储器和对数据对象建模的层（第10章）。

下面是本书各章节的主要内容：

第1章 谷歌App引擎简介

对在高层次上对谷歌App引擎及其组件、工具和主要特征进行了概述，并且从整体上介绍了谷歌云平台。

第2章 创建应用程序

本章为使用Java语言的简要教程，介绍了创建开发环境、使用模板引擎创建网页、创建账号和域名以及在App引擎上部署应用程序。教程中的应用程序展示了数个App引擎特征的使用—谷歌账号（Google Accounts）、数据存储和高速缓存，实现了许多网络应用程序共用的模式：存储和恢复用户首选项。

第3章 配置应用程序

本章描述了App引擎如何处理进来的请求，以及如何配置这些行为。本章介绍了App引擎的架构、前端的各种特征、App服务器和静态文件服务器。本章说明了前端如何将请求路由至App服务器和静态文件服务器，以及如何管理安全连接和谷歌账号认证和授权。本章还讨论了配额和限制，以及如何通过设置预算来对其进行提升。

第4章 请求处理器和实例

本章介绍了更深入地检测App引擎如何运行用户的代码。App引擎将进入的网络请求路由至请求处理器。请求处理器长期运行被称为实例的容器。App引擎创建并销毁实例，以适应用户流量的需求。用户可以通过编写线程安全代码并利用多线程特征更好地利用自己的实例。用户可以将自己的App架构组织成模块，每个模块可以单独地将实例集中起来，其中每个实例运行自己的代码和配置。

第5章 使用模块

模块的方式使得用户可以将自己的应用程序创建成部件的集合，其中每个部件都具有自己的扩展属性和性能特征。本章全面介绍模块相关内容，包括各种扩展选项、配置以及用户用于维护自己App模块的工具和API。

第6章 数据存储实体

本书使用了好几章的篇幅介绍云数据存储，本章是第1章。所谓云数据存储是指可扩展对象数据存储系统，其支持本地事务和一致性保证的两种模式（强模式和最终模式）。本章介绍了数据项、键和属性，以及从App引擎创建、更新和删除数据项的Java API。

第7章 数据存储查询

本章介绍了云数据存储查询和索引，以及用于查询的Java API。本章详细描述了查询引擎的特征，以及每个特征如何使用索引。本章还讨论了如何为自己的应用程序查询定义和管理索引。本章还涉及一些高级特征，如查询游标和投影查询。

第8章 数据存储事务

本章介绍如何使用事务保持数据的一致性。云数据存储可在可扩展环境中使用本地事务。用户App采用事务性单元的形式（被称为数据项群组）布置其数据项。本章全面地说明数据存储如何更新数据，以及如何设计用户数据和用户App才能更好地利用这些特征。

第9章 数据存储管理

本章介绍如何管理和发展用户App数据存储数据。云控制台、SDK工具体和管理API为用户数据提供了无数种视图以及与用户数据相关的信息（元数据和统计信息）。用户可以编写程序访问这些信息中的绝大部分，因此用户可以创建自己的管理程序。本章还讨论了如何使用远程API，即创建管理工具的代理，它运行在本地计算机上，但可以访问用户App可用的服务。

第10章 Java持久化API

本章简要介绍Java持久化API（JPA），如何将其概念转换成数据存储，如何使用它对数据模式建模，以及如何使用它让用户应用程序更易于与其他环境接口。JPA是Java EE的标准接口。App引擎也支持被称为Java数据对象（Java Data Objects，JDO）的其他标准的接口，但本书不介绍JDO的相关内容。本章还专门介绍了Java。

第11章 在App Engine中使用Google Cloud SQL

谷歌云SQL对MySQL数据库实例进行全面管理。用户可以将云SQL作为App引擎应用程序的关系数据库使用。本章完整地介绍了创建SQL实例、创建数据库、准备本地开发环境，并且从App引擎连接至云SQL。本章还讨论了云SQL的优秀特征（例如备份），以及导出和导入数据。云SQL对云数据存储和云存储进行了补充，是持久性存储的新选择，特别是需要关系数据库的情况下，云SQL是一个高性能的选择。

第12章 存储高速缓存

本章介绍App引擎的存储高速缓存服务（“存储高速缓存”）以及其Java API。积极高速缓存对于高性能网络应用程序是很重要的。

第13章 获取URL和网络资源

本章介绍如何通过使用URL获取（URL Fetch）服务经HTTP访问Internet上的其他资源。Java应用程序可以直接使用API以及通过Java标准库调用来调用这些服务。