

《Google App Engine开发》

图书基本信息

书名：《Google App Engine开发》

13位ISBN编号：9787564122683

10位ISBN编号：7564122684

出版时间：2010年6月

出版社：东南大学出版社

作者：Charles Severance

页数：241

版权说明：本站所提供下载的PDF图书仅提供预览和简介以及在线试读，请支持正版图书。

更多资源请访问：www.tushu000.com

前言

The greatest single reason that the World Wide Web has been so widely used and adopted is because individuals are allowed to participate in the Web. People can produce web Content and create a MySpace page or home pages provided by their school or organization and contribute their creativity and content to the Web. Free services like Blogger, Flickr, Google Sites, Google Groups, and others have given us all an outlet for our creativity and presence on the Web—at no charge. For most of the life of the Web, if you wanted to have your own rich software-backed website with data storage, your only choice was to purchase hosting services from an Internet Service Provider (ISP) and learn database management and a programming language like PHP to build or run your software. Learning and paying for this much technology was just beyond the reach of most web users, who simply had to accept the limited features of MySpace, Blogger, or whatever system hosted their web content. In April 2008, Google announced a product called App Engine. When you write a program for the Web that runs on App Engine, your software runs on the Google servers somewhere in the Google "cloud." It is as if you are a Google employee and you have access to the entire scalable Google infrastructure. App Engine captures much of Google's experience of building fast, reliable, and scalable websites, and through App Engine, Google is revealing many of the secrets about how its own applications scale to millions of users. The most exciting part of the Google App Engine announcement is the fact that it is free for moderate levels of use. Every person with a Gmail account can have a number of free applications running on the Google infrastructure. If your application becomes extremely popular and your traffic goes above the allowed levels of the free account, you can pay to use more of Google's resources. As your application scales, Google engineers and operations staff take care of all the hardware, data storage, backup, and network provisioning for you.

《Google App Engine开发贰》

内容概要

《Google App Engine开发(影印版)》内容简介：App Engine几乎是近年来最引人注目的Web技术，它提供了一个简单易用的应用程序框架以及基本的Web工具。尽管Google自带的教程已经展示了主要的步骤，但是《Google App Engine开发》将会帮助任何人开始上手使用这个平台。读完《Google App Engine开发(影印版)》你将学会如何构建完整的交互应用，并且将它们部署到和Google搜索引擎使用同样服务器的云中。

《Google App Engine开贰

作者简介

Charles Severance是密歇根大学信息学院的兼职助理教授。他也是IMS全球学习联合会的IMS开发者网络协调员。

《Google App Engine开贰

书籍目录

Preface1. Programming on the Web The Request/Response Cycle What Is Google App Engine ? What Is a "Cloud" ? Why Did Google Build App Engine and Give It Away for Free ? What Is the Google Infrastructure Cloud ? Enter the Application Engine Your Application Must Be a Good Citizen in the Google Cloud How the Cloud Runs Your Application Why You Really Want to Run in the Cloud The Simplest App Engine Application Summary Exercises2. HTMLandCSS A Brief History of HTML and CSS HyperText Markup Language (HTML) A Well-Formed HTML Document Validating Your HTML HyperText Links (Anchor Tags) Multiple Files Lists in HTML Specifying Styles Using CSS Styling Tags with CSS Exerting More Control over Markup Validating CSS Tools to Help You Work with css Building the Navigation Menu The CSS Box Model Adding Navigation Hints for Users Summary Exercises3. Python What Is Programming ? About Pyttion Installing Python The Essence of Programming Input, Processing, and Output Conditional Steps: A Number-Guessing Web Application Variables and Expressions Naming Variables Constants Handling Text Data Using Strings Using the String Library Types and Conversion Variables with Many Values at the Same Time: Lists Repeated Code: Loops Python's Backpack: Dictionaries Looping Through a Dictionary Stored and Reused Code: Python Functions Turning Traceback to Our Advantage Using Try and Except Object-Oriented Python Comments in Python The Tao of Programming Summary Exercises4. Sending Data to Your Application Making an HTTP Request Hacking the HTTP Protocol The HTTP Protocol Standards Watching HTTP in Action HTML: Building a Form Sending Form Data to the Server Using POST Handling Form Data in the App Engine Server Sending Form Data to the Server Using GET App Engine Log Looking at All the Data Available on an HTTP Request Advanced: Multipart POST Data with File Upload Summary Exercises5. TheApp Engine webapp Framework A Trivial App Engine Application An Equivalent Trivial webapp Application The Handler Callback Pattern Looking at the Handler Code What Is "self" and Why Is It There ? Number Guessing Using the webapp Framework Web Server Logs Summary Exercises6. Templates Template Syntax Using the Templates from Python The Number-Guessing Game Using Templates Abstraction and Separation of Concerns: "Model View Controller" Building a Multiscreen Application Static Files in App Engine Referencing Static Files Generalizing Template Lookup with Multiple Templates Extending Base Templates Conditional Processing Templates Replacing More Than One Block in the Base Template Extending Our Application Syntax Errors More on Templates Summary Exercises7. Cookies and Sessions HTTP Cookies Logging In and Logging Out Using Sessions Installing the Simple Session Utility Using the Session to Mark Login and Logout Changing the User Interface for a Logged-In User Summary Exercises8. AppEngineDatastore The Model-View-Controller Pattern Defining Models Adding a New Account Form Adding a User Account Looking Through the Datastore Logging In and Checking Accounts and Passwords Retrieving and Displaying Many Objects Terminology: Different Types of Keys References Between Data Models Interactions Between Models Pt/tting the Primary Key into Session Adding the Chat Feature Summary Exercises9. JavaScript, jQuery, and AJAX jquery Create a View of the Chat Messages in HTML Asynchronously Updating a

章节摘录

插图：Running an application in the cloud is kind of like flying business class across the Pacific Ocean between Australia and the United States. You are vaguely aware that you are going really fast inside of a highly complex device that you barely understand. The pilots, crew, maintenance people, chefs, logistics staff, traffic controllers, and gate agents all are making sure that your trip happens efficiently and comfortably—and that it is uneventful. All you know is that you sit in a recliner, watch a movie, eat a nice filet mignon, have a glass of red wine, lay the seat flat, sleep for a few hours, and wake up refreshed on a different continent.

Why You Really Want to Run in the Cloud

You might initially think that you don't want to run in the Google cloud because you want to make your own decisions and control your own destiny. You might want to run your own servers in your own facility and make all the decisions about your application. Perhaps you just like walking into a server room and seeing the hardware that is running the application. Although this sense of control might sound appealing at first, it is really just a lot of trouble and energy that does not advance the cause of your application. Here are a few of the things that you have to worry about when you run on your own servers: what operating system should I run? What version of the operating system is the most reliable? When do I apply vendor patches (especially those pesky security patches)? How do I protect my system from intruders? Do I need a firewall to protect my servers? How do I monitor my servers to detect when an intrusion happens and then how do I get notified? How far do I have to drive to the server room to reformat and reinstall the software at 4:00 a.m. so that it is back up by 10:00 a.m.? What database do I run? What version? What patches? Should I upgrade the memory of my database server, or should I add an additional disk to the RAID controller? Can I use a single database server, or do I need to cluster several database servers? How does the clustered database server get backed up? How long does it take to restore my database when there is a hardware problem with the database server's disk drives? How many application web servers do I need? I know that my application's peak usage is from 7:00 p.m. to 9:00 p.m. each day. Do I buy enough hardware to handle that peak load, or do I buy a little less hardware and just let the servers slow down a bit during the 7:00 p.m. to 9:00 p.m. period? If my application is so popular that it is used both in the United States and Europe, do I need to find a data center in Europe and put some hardware in Europe so that all the European users see a quick response time? When should I upgrade my hardware? Should I add more hardware and keep the old hardware or simply pitch the old hardware and install all new hardware? How much energy does my hardware take? Is there a way to reduce the energy footprint of my hardware?

《Google App Engine开发》

媒体关注与评论

“《Google App Engine开发》使几乎不可能的事成为可能，帮助我这个老家伙学习一些难以置信的相关技术诀窍。我在过去整个职业生涯中一直想要更好地掌握Web开发，自从有了这本书，梦想终于成真。” ——Robert Glushko . 学生，密歇根大学

《Google App Engine开贰

精彩短评

- 1、本来以为这本书能讲一些google app engine 官方文档中没有讲的内容。到手后才发现这本书定位是初学者。花了大量篇幅讲PYTHON,****协议, CSS等。而GAE方面则只讲了少量与构建网站有关的部分, 很多高级特性在本书中都没有提。最后这本书还花了大量篇幅将怎么安装GOOGLE APP ENGINE, 这些完全可以在google的文档中找到。
个人觉得, 如果对网站开发一窍不通, 可以考虑看看这本书。如果想深入学习GAE, 那最好还是看GOOGLE提供的文档和样例代码。
- 2、书还可以, 有点损伤, 还没看
- 3、一章html, 一章http, 一章css, 一章python, 后面四章怎么安装, 整本书不知所云, 一点没有深入, 上面的内容在google app engine首页就可以覆盖。把作者的twitter找出来骂一顿。

版权说明

本站所提供下载的PDF图书仅提供预览和简介，请支持正版图书。

更多资源请访问:www.tushu000.com