

<<拨得云开见日出>>

图书基本信息

书名：<<拨得云开见日出>>

13位ISBN编号：9787121180118

10位ISBN编号：7121180111

出版时间：2012-9

出版时间：电子工业出版社

作者：刘川意，袁玉宇 著

页数：561

字数：892000

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：<http://www.tushu007.com>

## &lt;&lt;拨得云开见日出&gt;&gt;

## 前言

我们需要说明的第一句话是：这本书主要是写给“要动手”的人们的！

如何构建一个云计算系统，这是从事云计算相关工作的程序员和工程师最关心的问题之一。但是很遗憾的是，市面上的绝大多数书籍主要介绍的是云计算的模式、架构、应用、接口等，无法使读者站在一个云计算系统的内部，清楚地获知一个典型的云计算系统是如何构成，以及如何编码实现的。

本书以一个典型的开源基础设施云Eucalyptus为例，解构了一个典型的云计算系统的主要构成模块，并在源代码层面上详细讲解了如何构建一个典型的云计算系统。

更进一步的，本书通过动手实验的方式，使读者能直观地掌握如何编译一个云系统，其代码结构是怎么样的；并通过写测试代码，使读者掌握如何去调用云系统的API，如何添加和定制化功能。

通过这本书，我们希望读者有以下的收获：如何根据需求选择合适的云计算系统？

随着云计算模式的逐渐普及和广泛使用，可以想象，越来越多的企业、机构和个人用户主要面临的问题不再是如何采购硬件、软件，再组织人或外包进行系统开发，然后进行系统部署和运维。

他们经常要做的事情是根据应用的需求，选择合适的云计算系统来承载相应的应用或选择合适的云集算系统部署自己的私有云。

本书在对工业界和学术界的主流云计算系统进行广泛调研的基础上，提出了典型的云集算系统体系结构，并以目前代表性的开源云系统为例，对主要子系统、关键功能和性能参数、核心技术等进行了系统的介绍和对比分析，从而为读者分析和判断市场上的云平台提供了系统的评价方法。

更进一步的，本书在各子系统的相应章节，通过使用该子系统的接口API，定量测试和评价云平台的各项指标，以及示例如何将云平台与用户的应用平台进行对接。

从源代码层面，一个典型的云计算系统是如何构建和实现的？

我们可以发现一个普遍存在的事实：绝大多数的云计算系统，包括商业云平台，如Amazon，Microsoft Azure等，其物理服务器的主流配置是Linux或Unix类操作系统。

而我们知道，在这类操作系统中，有大量优秀的产品级开源系统或工具可供使用和集成，如何将这些开源系统和工具用于构建云平台？

本书分别针对云计算系统的主要核心子系统，如虚拟机管理器、存储网络软件、模块互连框架等，详细介绍了如何使用这些工具，并在此基础上实现云计算系统的核心模块。

如何实现一个产品级的云计算系统？

如何寻找典型的云计算系统中有关键技术点？

本书分析的不是玩具系统或原型系统，而是真实应用于工业生产的产品系统！

本书选择的Eucalyptus私有云系统已经被世界500强中25%的企业所部署和使用。

读者以此为基础，可以实现和构建具有商业意义和价值的关键技术点与功能点。

最后，要特别感谢北京邮电大学可信分布式计算与服务教育部重点实验室2009级和2010级的同学们，没有你们的辛苦工作和努力付出，这本书的出版将没有可能。

在此请允许一一列举你们的名字：2009级的宋小宝、李思扬、唐博、王珂、王彦丞、万磊、张路、陈晏民同学；2010级的李春文、刘彬、刘潇健、王小亮、颜夏青同学。

这本书是大家共同奋斗，并肩合作的见证！

著者 2012年9月于北京邮电大学

## <<拨得云开见日出>>

### 内容概要

如何构建一个云计算系统是广大的程序员和软件工程师最关心的问题之一。但是很遗憾的是，市面上的绝大多数书籍主要介绍的是云计算的模式、架构、应用、接口等，无法使读者站在一个云计算系统的内部，清楚地获知一个典型的云计算系统是如何构成，以及如何编码实现的。

本书从基础设施云入手，通过对当前主要的开源基础设施云进行分析，总结了一个典型的云计算系统的组织结构，并分别介绍了虚拟化运行环境子系统、云存储子系统、虚拟网络子系统、模块互连子系统等主要的子系统，其体系结构、代码结构、主要的实现模块以及关键的代码。

最后对每一个相应的子系统，通过实际将其编译、运行、测试，使读者能对云系统能在实践中进行解构和修改。

## <<拨得云开见日出>>

### 书籍目录

#### 第1章 引言

- 1.1 本书的特点与结构
- 1.2 基础设施云介绍

#### 第2章 基础设施云对比分析

- 2.1 基础设施云对比分析概述
- 2.2 典型基础设施云的构成

#### 第3章 虚拟化运行环境

- 3.1 概述
- 3.2 体系结构
- 3.3 相关技术
- 3.4 资源管理
- 3.5 节点调度
- 3.6 虚拟机生命周期管理
- 3.7 虚拟机监控
- 3.8 开发环境搭建
- 3.9 单元测试及开发实例
- 3.10 下一步工作

#### 第4章 虚拟磁盘系统

- 4.1 概述
- 4.2 体系结构
- 4.3 相关技术
- 4.4 代码结构
- 4.5 元数据管理
- 4.6 虚拟磁盘生命周期管理
- 4.7 存储介质管理
- 4.8 连接管理
- 4.9 开发环境搭建
- 4.10 单元测试及开发示例
- 4.11 下一步工作

#### 第5章 基于Web的在线云存储系统——Walrus

- 5.1 概述
- 5.2 Walrus体系结构
- 5.3 Walrus相关技术
- 5.4 Walrus元数据管理
- 5.5 Walrus 代码结构
- 5.6 Walrus对用户文件夹的管理
- 5.7 Walrus对用户文件的管理
- 5.8 对镜像文件的管理
- 5.9 Walrus对快照文件的管理
- 5.10 Walrus开发环境搭建
- 5.11 Walrus使用方法
- 5.12 Walrus待解决的问题

#### 第6章 镜像管理

- 6.1 操作系统安装及基本设置
- 6.2 镜像系统的更新

<<拨得云开见日出>>

6.3 “镜像超市”

6.4 客户端与虚拟运行环境的数据传输

6.5 下一步工作

第7章 虚拟网络系统

第8章 接入与互连系统

## &lt;&lt;拨得云开见日出&gt;&gt;

## 章节摘录

版权页：插图：4.3.2 LVM2命令 LVM是Logical Volume Manager（逻辑卷管理）的简称，它是Linux环境下对磁盘分区进行管理的一种机制，LVM是建立在硬盘和分区之上的一个逻辑层，为文件系统屏蔽下层磁盘分区布局，从而提高磁盘分区管理的灵活。

LVM可以将不问种类的物理卷组织成卷组，可以随时通过增加或减少物理卷的方式来改变卷组的大小。

用户可以在卷组之上创建自己所需要的逻辑卷，同时也可以增大或减小逻辑卷的存储中间。

LVM2是LVM的第2版，它为Linux提供逻辑卷管理功能，并完全向后兼容原来的LVM工具集。

LVM基本术语如下。

（1）物理卷（Physical Volume, PV） 物理卷是LVM系统中的最低层；物理卷是LVM的基本存储

逻辑块：它可以是整个硬盘、硬盘上的分区或从逻辑上与磁盘分区具有同样功能的设备（如RAID）。

（2）卷组（Volume Group, VG） 卷组建立在物理卷之上，它由一个或多个物理卷组成；在卷组之上可以创建一个或多个逻辑卷；卷组的大小可以通过增加或减少逻辑卷的方式改变。

（3）逻辑卷（Logical Volume, LV） 逻辑卷建立在卷组之上；逻辑卷的大小可以动态调整；逻辑卷类似于非LVM系统的硬盘分区，是用户真正使用的存储区域。

（4）物理区域（Physical Extent, PE） PE是物理卷的基本单元；PE是可被LVM系统寻址的最小存储单元；PE大小可以指定，默认为4MB；同一卷组中物理卷的PE大小一致。

（5）逻辑区域（Logical Extent, LE） LE是逻辑卷的基本单元；在同一卷组中，LE和PE是一一对应的；LVM系统创建LE并自动维护PE / LE对应表，使得每一个LV里面的LE都可以找到与之对应的PE。

LVM系统中，PV、VG、LV关系如图4.3所示。

LVM的相关命令如下。

（1）物理卷相关命令 Pvcreeate：创建物理卷。

Pvscan：在系统的所有磁盘中搜索已存在的物理卷。

Pvdisplay：显示物理卷信息。

Pvchange：改变物理卷的分配许可设置。

Pvmove：把某物理卷中的数据转移到同卷组中其他的物理卷中。

Pvremove：删除物理卷。

（2）卷组命令 Vgcreate：创建卷组。

Vgdisplay：显示卷组信息。

Vgchange：改变卷组属性。

Vgextend：添加一个物理卷至卷组。

Vgreduce：从卷组中删除物理卷。

Vgscan：扫描所有磁盘寻找逻辑卷组。

Vgremove：删除卷组。

Vgexport：用于导出卷组。

待输出卷组必须为非激活状态。

卷组中的物理卷需要被转移到其它系统中使用时，就需要使用这个命令。

## <<拨得云开见日出>>

### 编辑推荐

《拨得云开见日出：解构一个典型的云计算系统》可作为工科院校相关专业师生的教材或辅导材料，也可供云计算相关研发人员和工程人员作为参考材料。

<<拨得云开见日出>>

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:<http://www.tushu007.com>