

▶▶ 高等学校应用型本科“十三五”规划教材



- 云计算专业入门级教材
- 结合OpenStack的部署过程进行讲解

OpenStack 技术原理与实战

主 编 韩 璞
副主编 陈 可
主 审 刘黎明

高等学校应用型本科“十三五”规划教材

OpenStack 技术原理与实战

主 编 韩 璞

副主编 陈 可

主 审 刘黎明

西安电子科技大学出版社

内 容 简 介

本书结合 OpenStack 整体架构,对 OpenStack 云平台核心组件的工作原理进行介绍与分析,并通过 OpenStack 的实践部署,将 OpenStack 的理论与实践相结合,使读者能够在了解 OpenStack 部署和安装的同时,熟悉 OpenStack 内部核心组件的协作关系。

全书在组织形式上,采用理论与实践相结合的描述方式,图文并茂地将 OpenStack 具体的理论知识形象地呈现给读者,并通过具体的配置案例,引导读者将每一个组件部署在 OpenStack 的云平台上。

本书适合应用型本科院校软件工程专业及计算机类其他专业云计算类课程使用,也可供对云平台部署有兴趣的其他读者使用。

图书在版编目(CIP)数据

OpenStack 技术原理与实战/韩璞主编. —西安:西安电子科技大学出版社,2016.4

高等学校应用型本科“十三五”规划教材

ISBN 978-7-5606-4045-7

I. ① O… II. ① 韩… III. ① 计算机网络—高等学校—教材 IV. ① TP393

中国版本图书馆 CIP 数据核字(2016)第 043393 号

策 划 李惠萍 戚文艳

责任编辑 李惠萍 张 欣

出版发行 西安电子科技大学出版社(西安市太白南路 2 号)

电 话 (029)88242885 88201467 邮 编 710071

网 址 www.xduph.com 电子邮箱 xdupfb001@163.com

经 销 新华书店

印刷单位 陕西天意印务有限责任公司

版 次 2016 年 4 月第 1 版 2016 年 4 月第 1 次印刷

开 本 787 毫米×1092 毫米 1/16 印 张 11.5

字 数 265 千字

印 数 1~3000 册

定 价 22.00 元

ISBN 978-7-5606-4045-7/TP

XDUP 4337001-1

如有印装问题可调换

本社图书封面为激光防伪覆膜,谨防盗版。

西安电子科技大学出版社

高等学校应用型本科“十三五”规划教材

编审专家委员会名单

主任：鲍吉龙（宁波工程学院副院长、教授）

副主任：彭 军（重庆科技学院电气与信息工程学院院长、教授）

张国云（湖南理工学院信息与通信工程学院院长、教授）

刘黎明（南阳理工学院软件学院院长、教授）

庞兴华（南阳理工学院机械与汽车工程学院副院长、教授）

电子与通信组

组长：彭 军（兼）

张国云（兼）

成员：（成员按姓氏笔画排列）

王天宝（成都信息工程学院通信学院院长、教授）

安 鹏（宁波工程学院电子与信息工程学院副院长、副教授）

朱清慧（南阳理工学院电子与电气工程学院副院长、教授）

沈汉鑫（厦门理工学院光电与通信工程学院副院长、副教授）

苏世栋（运城学院物理与电子工程系副主任、副教授）

杨光松（集美大学信息工程学院副院长、教授）

钮王杰（运城学院机电工程系副主任、副教授）

唐德东（重庆科技学院电气与信息工程学院副院长、教授）

谢 东（重庆科技学院电气与信息工程学院自动化系主任、教授）

楼建明（宁波工程学院电子与信息工程学院副院长、副教授）

湛腾西（湖南理工学院信息与通信工程学院教授）

计算机大组

组 长：刘黎明（兼）

成 员：（成员按姓氏笔画排列）

刘克成（南阳理工学院计算机学院院长、教授）

毕如田（山西农业大学资源环境学院副院长、教授）

向 毅（重庆科技学院电气与信息工程学院院长助理、教授）

李富忠（山西农业大学软件学院院长、教授）

张晓民（南阳理工学院软件学院副院长、副教授）

何明星（西华大学数学与计算机学院院长、教授）

范剑波（宁波工程学院理学院副院长、教授）

赵润林（山西运城学院计算机科学与技术系副主任、副教授）

黑新宏（西安理工大学计算机学院副院长、教授）

雷 亮（重庆科技学院电气与信息工程学院计算机系主任、副教授）

前 言

随着云计算技术的日趋流行，云计算已经成为一个与我们息息相关的应用产品。这种日渐流行的 IT 技术，正推动着这个行业的革命性变化和第三次 IT 浪潮。当下一些完备的云计算商业产业链也逐渐形成，例如亚马逊的 EC2、VMware 公司的一系列产品等。这些使得云计算不仅成为一项优秀的 IT 技术，也逐渐成为一种新的商业计算模型和 IT 服务运营模式，特别是在移动互联网日渐成熟的今天，云计算使人们“像使用自家的水、电一样”方便快捷地使用运营商提供的任何形式的计算、网络等资源，而不需要在这些硬件等基础设施上增加投入。

在诸多云计算相关的产品中，云平台是一种相对典型且成熟的云产品。它采用云计算三种模式中的基础设施即服务(IaaS)模式，能够灵活地配置用户需要的计算资源等基础设施，用户能够按需使用云平台上的一切虚拟资源。OpenStack 是由 NASA(美国国家航空航天局)和 Rackspace 合作研发并发起的一个开源的云计算管理平台项目，它是 IaaS 云计算解决方案。它通过使用 KVM 等虚拟化技术，将服务器的硬件进行虚拟，根据用户的需求可以随意配置，从而能够对外提供强大的计算能力。用户通过网络可以使用 OpenStack 平台中的虚拟计算机，平台管理员可以通过后台或管理页面进行整个云平台资源的管理和配置。

然而，由于 OpenStack 的部署是一个较为繁琐的过程，其本身所包含的组件是以插件的形式进行组合之后部署在 OpenStack 的计算节点和控制节点上的，对于初学者而言完成这一阶段的学习较为困难。本书针对 OpenStack 架构进行深入的分析，对 OpenStack 组件的构成及协作流程进行介绍，从 G 版 OpenStack 的各个组件的工作原理出发，介绍不同组件的作用及工作过程。书中以 G 版 OpenStack 的部署过程为分析案例，以“先理论，再实践”的思路，从第二章 OpenStack 整体系统架构开始，每一章中均包含 OpenStack 每一组件的部署和安装过程的相关内容，同时还配有比较详细的原理介绍。

本书主要适用于 OpenStack 初学者理解与认识云平台技术，从理论和技术上，培养学员部署 OpenStack 的实践能力，在实践中提高学员对理论的理解与认识，培养初学者的工程部署经验和习惯，从而使其能够进行云计算其他领域的技术使用与开发。

本书内容主要涵盖 OpenStack 核心组件的工作原理与部署安装，为了遵循“教、学、做”一体化教学模式，在每章内容的编排上，遵循“学以致用，理论结合实践”的原则，以培养学生实践能力为目标，在保证对 OpenStack 基本理论的认知的基础上，注重 OpenStack 工程实践中的配置、安装。

本书共九章，第一章是云计算与 OpenStack 简介，第二章是 OpenStack 整体系统架构，第三章是 Nova 组件，第四章是 keystone 认证组件，第五章是 Glance 镜像组件，第六章是 Storage 分布式存储组件，第七章是 Quantum 网络组件，第八章是 Horizon 前端界面组件，第九章是 OpenStack 部署与调试。在前两章的学习过程中，通过云计算与云平台的基本概念和 OpenStack 的基本构成，详细介绍了 OpenStack 云平台的整体结构，使读者对 OpenStack 具有一个初步的整体认识；在后续的五章中，针对 OpenStack 的计算组件 Nova、认证组件

keystone、镜像组件 Glance、存储组件 Swift 和 Cinder、网络组件 Quantum 以及前端界面组件 Horizon 进行介绍，特别是在对每个组件的介绍过程中，首先从原理上对 OpenStack 的各个核心组件进行分析，然后通过具体的配置和部署，介绍相关理论基础，最后通过完整的部署和调试指导将本书的内容进行系统的总结和归纳。

本书由刘黎明教授主审，并负责本书的统筹规划，韩璞任主编，负责全书统稿，陈可为副主编。其中，第二章、第四章、第五章、第七章、第八章和第九章由韩璞编写；第一章、第三章和第六章由陈可编写。

本书在编写过程中得到了各编委的大力支持，同时，同行专家及相关行业人士也提出了很多宝贵意见，在此表示感谢。

最后，由于 OpenStack 为版本升级和发展较快的开源云平台，尽管编委会成员在编写过程中付出了很多，但限于编者的水平和时间仓促，欠妥之处在所难免，读者如有宝贵意见和建议，可随时联系我们(我们的邮箱为 hanp2008@126.com)，我们会积极吸取您的建议，同时我们也会时刻关注 OpenStack 新版本中的新技术和发展方向，这些将会在本书的后续版本中及时体现和修改。

作者

2016年1月

目 录

第一章 云计算与 OpenStack 简介 1	
1.1 云计算的概念..... 1	
1.2 云计算工作模式..... 2	
1.3 云计算的特点..... 3	
1.4 云计算的服务模式..... 3	
1.5 OpenStack 概述..... 4	
1.6 OpenStack 的功能与作用..... 5	
1.7 OpenStack 部署方式介绍..... 7	
1.8 OpenStack 的应用现状与发展趋势..... 8	
1.8.1 OpenStack 的版本演变..... 8	
1.8.2 OpenStack 的发展趋势..... 10	
1.9 其他开源云平台简述..... 10	
1.9.1 Eucalyptus..... 10	
1.9.2 AbiCloud..... 12	
1.9.3 OpenNebula..... 12	
第二章 OpenStack 整体系统架构 14	
2.1 OpenStack 基本框架..... 14	
2.1.1 OpenStack 核心组件..... 14	
2.1.2 OpenStack 架构的设计原则..... 15	
2.2 OpenStack 整体架构模型..... 16	
2.2.1 OpenStack 的功能构成..... 16	
2.2.2 OpenStack 逻辑结构与模型..... 17	
2.3 OpenStack 物理架构..... 19	
2.4 OpenStack 的运行机制与流程..... 20	
2.4.1 AMQP 消息处理与响应机制..... 20	
2.4.2 OpenStack 工作流程..... 22	
2.4.3 OpenStack 平台管理流程..... 23	
2.5 完善中的 OpenStack..... 24	
2.6 OpenStack 部署准备..... 25	
第三章 Nova 组件 26	
3.1 Nova 的基本概念..... 27	
3.2 Nova 工作原理及组件构成..... 30	
3.2.1 Nova 核心组件的构成..... 30	
3.2.2 Nova 组件的协作关系..... 32	
3.3 Nova-API 模块..... 33	
3.3.1 Nova API 的作用..... 34	
3.3.2 Nova API 中的 WSGI 接口..... 35	
3.3.3 Nova API 结构..... 35	
3.3.4 Nova API 服务流程..... 36	
3.3.5 扩展 API(Extension API)..... 40	
3.4 Nova-Client 功能分析..... 43	
3.5 Nova-Compute 模块..... 44	
3.6 Nova 中的 RabbitMQ 解析..... 45	
3.6.1 RabbitMQ..... 46	
3.6.2 AMQP..... 46	
3.6.3 RabbitMQ 在 Nova 中的实现..... 47	
3.6.4 rpc.call 和 rpc.cast 调用流程..... 51	
3.7 Nova-Schedule 模块..... 52	
3.8 Nova-Cell 模块..... 54	
3.8.1 Nova Cell 模块简介..... 54	
3.8.2 Nova Cell 模块基础架构..... 54	
3.8.3 Nova Cell 模块中主要组件介绍..... 55	
3.8.4 Nova Cell 环境配置与搭建..... 56	
3.9 Nova 的安装与配置..... 59	
3.9.1 创建数据库..... 60	
3.9.2 配置..... 60	
3.9.3 启动服务..... 62	
3.9.4 同步数据并启动服务..... 62	
3.9.5 查看服务..... 62	
3.9.6 组策略..... 62	
3.9.7 检查故障..... 63	
第四章 keystone 认证组件 64	
4.1 认识 keystone..... 64	
4.2 keystone 架构..... 65	
4.3 keystone 的基本概念和数据模型..... 65	
4.3.1 keystone 基本概念..... 65	
4.3.2 角色关联..... 67	
4.3.3 keystone 数据模型..... 68	
4.4 keystone 的工作原理..... 71	
4.5 安装与配置 keystone..... 73	

4.5.1 安装 Keystone 的准备工作	73	6.3.1 配置要求	119
4.5.2 keystone 相关的配置文件	75	6.3.2 安装过程	119
4.5.3 keystone 安装过程	75	第七章 Quantum 网络组件	123
4.5.4 keystone 安装验证	79	7.1 Quantum 概述	123
第五章 Glance 镜像组件	82	7.1.1 OpenStack 网络基本构成	123
5.1 Glance 概述	82	7.1.2 Quantum 基本概念	125
5.1.1 Glance 功能	82	7.2 Quantum 架构	127
5.1.2 Glance 基本概念	82	7.2.1 Quantum 网络架构	127
5.2 Glance 架构与数据模型	84	7.2.2 Quantum 网络原理	128
5.2.1 Glance 逻辑架构	85	7.2.3 Quantum 逻辑模型	129
5.2.2 Glance 数据库设计	86	7.3 OpenStack 网络模式	131
5.3 Glance 的关键配置文件	88	7.3.1 Flat 模式	131
5.3.1 glance-api.conf 文件	89	7.3.2 FlatDHCP 模式	132
5.3.2 glance-registry.conf 文件	89	7.3.3 VLAN 模式	132
5.3.3 其他配置文件	90	7.4 OpenStack 网络通信原理	133
5.4 Glance 的后端存储与工作流程	90	7.4.1 OpenStack 中的固定 IP 和 浮动 IP	133
5.4.1 镜像的后端存储	91	7.4.2 Quantum 通信流程	134
5.4.2 Glance 组件的工作流程	91	7.5 Quantum 的安装与部署	136
5.5 Glance 安装与部署	92	7.5.1 准备工作	137
5.5.1 准备工作	92	7.5.2 Quantum 的安装	137
5.5.2 Glance 安装过程	94	第八章 Horizon 前端界面组件	140
5.5.3 验证 Glance 的安装	96	8.1 Horizon 组件概述	140
5.6 镜像制作	98	8.1.1 Horizon	140
5.6.1 Ubuntu 镜像的制作	98	8.1.2 Horizon 功能	141
5.6.2 Windows 镜像的制作	99	8.2 Horizon 基本架构	141
第六章 Storage 分布式存储组件	101	8.2.1 Django	141
6.1 Swift 对象存储	102	8.2.2 Horizon 架构	142
6.1.1 基本原理	102	8.3 Horizon 工作原理和定制	143
6.1.2 Swift 架构及主要组件	104	8.3.1 Horizon 的工作过程	143
6.1.3 Swift 特性	108	8.3.2 定制 Horizon	144
6.1.4 应用场景	108	8.4 Horizon 安装与部署	144
6.2 Cinder 块存储	109	8.4.1 准备工作	144
6.2.1 Cinder 架构	111	8.4.2 安装 Horizon	145
6.2.2 Cinder 服务	112	8.5 Horizon 中的 Openstack	145
6.2.3 Cinder 插件	114	8.5.1 登录界面	145
6.2.4 Cinder 操作	115	8.5.2 云平台资源管理	147
6.2.5 Cinder 支持典型存储	116	8.5.3 管理镜像	147
6.2.6 Cinder 在 IT 环境中的主要问题	118	8.5.4 管理虚拟机	148
6.3 安装与配置 Cinder	119		

8.5.5 管理用户	150	9.1.9 Horizon 安装	165
第 9 章 OpenStack 部署与调试	152	9.2 问题与调试	165
9.1 OpenStack 部署	152	9.2.1 日志文件	165
9.1.1 系统准备	152	9.2.2 常见错误	166
9.1.2 网络基本配置	153	9.3 基于 OpenStack 的实践案例	169
9.1.3 基础组件安装	154	9.3.1 FastCloud 云平台	169
9.1.4 keystone 安装	154	9.3.2 FastCloud 应用场景	170
9.1.5 Glance 安装	157	9.3.3 案例系统实现	171
9.1.6 Quantum 安装	158	9.3.4 案例系统演示	172
9.1.7 Nova 安装	160	9.4 OpenStack 的未来	174
9.1.8 Cinder 安装	163		

第一章 云计算与 OpenStack 简介

1.1 云计算的概念

2006年8月9日, Google 首席执行官埃里克·施密特(Eric Schmidt)在搜索引擎大会(SES San Jose 2006)上首次提出“云计算”(Cloud Computing)的概念。Google 的“云计算”概念源于 Google 工程师克里斯托弗·比希利亚所做的“Google 101”项目。2007年10月, Google 与 IBM 开始在美国大学校园(包括卡内基梅隆大学、麻省理工学院、斯坦福大学、加州大学伯克利分校及马里兰大学等)推广云计算计划, 期望通过这项计划有效地降低分布式计算技术在学术研究方面的成本, 该计划为这些大学提供相关的软硬件设备及技术支持(包括数百台个人电脑及相应的服务器, 这些计算平台将提供 1600 个处理器, 支持包括 Linux、Xen、Hadoop 等开源系统)。

云计算自诞生之日起, 并没有准确而统一的定义。本书通过狭义和广义两方面对云计算的概念进行了一些阐述。狭义的云计算是指 IT 基础设施的交付和使用模式, 具体是指用户通过网络以按需、易扩展的方式获得所需的资源。提供资源的网络被称为“云”, “云”中的资源在使用者看来是可以无限扩展的, 并且可以随时获取, 按需使用, 随时扩展, 按使用量付费。由于“云”中的资源具有这些特性, 使得我们可以像使用水电一样使用 IT 基础设施。广义的云计算是指 IT 服务的交付和使用模式, 具体是指用户通过网络以按需、易扩展的方式获得所需的服务。这种服务可以是和软件、互联网相关的服务, 也可以是任意其他的服务, 它具有超大规模、虚拟化、可靠安全等特性。

在云计算定义中的“云”是指一些可以自我维护 and 管理的虚拟计算资源, 通常为一些大型服务器集群, 包括计算服务器、存储服务器、宽带资源等。云计算将所有的计算资源集中起来, 并由软件实现自动管理, 无需人为参与。这使得应用提供者无需为繁琐的细节而烦恼, 能够更加专注于自己的业务, 有利于创新和降低成本。这就好比是从古老的单台发电机模式转向了电厂集中供电的模式, 它意味着计算能力也可以作为一种商品进行流通, 就像煤气、水电一样, 取用方便, 费用低廉。与电厂集中供电相比, 云计算的最大不同在于它的服务是通过互联网进行传输的。

通过云计算的定义可以看出, 云计算的提出旨在使个人计算机的性能最小化, 功能最大化。用户只需用互联网连接设备访问云计算所提供的服务即可随时随地使用云平台所提供的各种资源。云计算技术是 IT 产业界的一场技术革命, 已经成为了 IT 行业未来发展的方向。各国政府纷纷将云计算服务视为国家软件产业发展的新机遇。例如, 近年来美国政府在 IT 政策和战略中也加入了云计算因素。美国国防信息系统部门(DISA)正在其数据中心

内部搭建云环境，早在 2009 年 9 月 15 日，美国总统奥巴马宣布将执行一项影响深远的长期性云计算政策，希望借助应用虚拟化技术来压缩美国政府支出；中国政府在“十二五”信息规划的技术背景中对云计算技术也做了阐述，明确提出云计算技术是我国下一个五年信息化产业发展的重点领域之一。在 2009 年年末，中国工业和信息化部出台的扶持软件产业发展指导意见中就明确了软件行业的 10 个发展重点，分别是基础软件、信息安全软件、工业软件、嵌入式软件、行业应用解决方案、系统集成和支持服务、软件服务外包、各类创新型服务、数字内容加工处理与服务以及 IC 设计服务。随着信息化程度的提高，IT 能耗急剧增加，而云技术的应用方法在环保与节能方面可以发挥巨大的潜能。美国环境保护局在 2010 年 8 月发布的一份研究报告中称，2009 年全美国的数据中心和服务器消耗了大约 6.1×10^{10} kW·h 的电能，企业 IT 数据中心的能耗已经占据美国总耗电量的 1.5%，相当于 580 万户普通美国家庭的日常用电，换算过来就是每年 45 亿美元。而这份报告分析认为，这些数字到 2016 年将几乎翻倍。同样，早在 2010 年，中国 PC 的社会保有量已经超过 1 亿台，服务器超过 200 万台，还有数目众多的路由器、交换机等其他 IT 设备，这些总量惊人的 IT 设备大约会消耗 $3 \times 10^{10} \sim 5 \times 10^{10}$ 的电能，等同于向大气中排放上千万吨的温室气体。云计算作为一种低功耗的按需分配技术，受到了各个追求低碳环保国家的极大重视。

1.2 云计算工作模式

在云计算模式中，用户通过终端设备接入网络，向“云”提出请求；“云”接受请求后组织资源，通过网络为“端”提供服务。通过这种模式，用户终端的功能可以大大简化，诸多复杂的计算与处理过程都将转移到终端背后的“云”上去完成。在整个过程中，用户所需的应用程序不再运行在用户的个人电脑、手机等终端设备上，而是运行在互联网的大规模服务器集群中；用户所处理的数据也无需存储在本地，而是保存在互联网上的数据中心里。云计算服务供应商负责这些数据中心和集群服务器正常运转的管理和维护，并保证为用户提供足够强的计算能力和足够大的存储空间。用户只要能够连接至互联网，在任何时间和任何地点，都可以访问云，实现按需随用。

从技术层面上讲，云计算是随着处理器技术、虚拟化技术、分布式存储技术、宽带互联网技术和自动化管理技术的发展而产生的。其基本功能的实现取决于两个关键因素：一个是数据的存储能力，另一个是分布式的计算能力。因此，云计算中的“云”可以再细分为“存储云”和“计算云”，也即“云计算 = 存储云 + 计算云”。存储云的产生依托于大规模的分布式存储系统，分布式存储系统的作用是将数据分散存储在多台独立的设备上，利用多台存储服务器分担存储负荷。计算云则依托于虚拟化技术和并行计算技术。虚拟化最主要的意义是用更少的资源做更多的事。在计算云中引入虚拟化技术，就是力求在较少的服务器上运行更多的并行计算，对云计算中所应用到的资源进行快速而优化的配置等。而并行计算的作用是首先将大型的计算任务拆分，然后再派发到云中的节点上进行分布式并行计算，最终将结果收集后统一整理，如排序、合并等。

1.3 云计算的特点

随着互联网的发展,云计算在日常生活中的应用已非常普遍。云计算是一种计算模型和互联网运营模式,它将诸如运算能力、存储、网络 and 软件等资源抽象成为服务,以便让用户通过互联网远程享用,同时付费用户也如同使用传统公共服务设施一样,进行有偿使用。因而,“因需而定”、“提供方便”、“动态改变”和“无限虚拟化的扩展能力”是云计算的几个重要特征。

依据云计算的几个重要特征,云计算系统需提供的五大核心特点如下所述。

(1) 按需自助服务(On Demand Self-Service)。供应商的资源保持高可用和高就绪的状态,用户可以按需方便、自助地获得资源。

(2) 泛在的网络访问(Broad Network Access)。用户可以通过网络,使用不同的终端设备(例如手机、笔记本电脑等)以统一的标准模式获取服务提供商的服务。

(3) 动态的资源池(Resource Pooling)。供应商的计算资源可以被整合为一个动态资源池,以多租户模式服务所有客户,不同的物理和虚拟资源可根据客户需求动态分配。服务提供商需实现资源的位置无关性,客户一般在不需要知道所使用的资源的确切地理位置的情况下,使用资源池中的资源。

(4) 快速弹性的资源提供方式(Rapid Elasticity)。服务提供商可以迅速、弹性地提供服务,能快速扩展,也可以快速释放,还可以实现快速缩小。对客户来说,可以租用的资源看起来似乎是无限的,可在任何时间购买任何数量的资源。

(5) 可计量的服务(Measured Service)。服务的收费可以是基于计量的一次一付,或基于广告的收费模式。云计算系统以针对不同服务需求(例如,CPU 时间、存储空间、带宽、用户账号的使用率高低)来计量资源的使用情况和定价,以提高资源的管控能力和促进资源的优化利用。整个系统资源可以通过监控和报表的方式使其对服务提供者 and 使用者透明化。

1.4 云计算的服务模式

云计算的三个服务模式(Delivery Models)是: SaaS(Software as a Service)、PaaS(Platform as a Service)和 IaaS(Infrastructure as a Service)。

(1) SaaS(软件即服务): 提供给客户的服务是运营商运行在云计算基础设施上的应用程序,客户可以在各种设备上通过瘦客户端界面访问,如浏览器。客户不需要管理或控制任何云计算基础设施,包括网络、服务器、操作系统、存储器等。

(2) PaaS(平台即服务): 提供给客户的服务是把客户用开发语言(例如 Java、python、.Net 等)和工具开发的应用程序部署到供应商的云计算基础设施上。客户不需要管理或控制底层的云基础设施,包括网络、服务器、操作系统、存储器等,但客户能控制部署的应用程序,也可以控制运行应用程序的托管环境配置。

(3) IaaS(基础设施及服务): 提供给客户的服务是客户对所有云计算设施的利用权限,

包括处理、存储、网络和其他基本的计算资源，客户能够部署和运行任意软件，包括操作系统和应用程序。客户不需要管理或控制任何云计算基础设施，但能控制操作系统的选择、储存空间、部署的应用，也可以获得有限制的网络组件(例如防火墙、负载均衡器等)的控制权。

总之，云计算的本质源于“服务”。在云计算的语境中，一个服务意味着一种可按需取用的状态。所以 SaaS 就意味着软件可以按需取用，它的关注点在于其内部的可用功能而不是应用之外的东西。PaaS 提供的是一种按需取用的正常运行环境，因此问题就成了把什么样的按需应用功能组合部署到这一环境中。由于正常运行环境是可以按需取用的，所以—一个部署到该环境中的应用也可以在按需取用的状态下运行。也就是说，这些部署到 PaaS 环境中的应用是可以按需交付的，结果就和 SaaS 一样。IaaS 指的是可以按需取用、按需预配置的基础设施。对 IT 专业人士来说，在运营层面预配置基础设施等同于部署服务器。而在云计算环境中，所有服务器都已虚拟化，而且是以虚拟机的形式部署的，所以 IaaS 最终就成了按需部署虚拟机的能力。

随着云计算创新的步伐不断加快，新一代的技术和成果也在快速增长。但是云计算市场的分散性导致客户难以选择云计算厂商和合作伙伴，一旦做错决定将不得不转移到新的云上进行重新构建。对于一些大的公司来说，这确实是一个挑战。鉴于上述原因，云需要一个开源的操作系统，开源云可以避免被锁的问题，而 OpenStack 就是这样一个开源的云操作系统。

OpenStack 是 IaaS 平台，它可以让任何人自行建立和提供云端运算服务。OpenStack 如同 Linux 一样，旨在构建一个内核，所有的软件厂商都围绕着它进行工作。从组件构成来看，OpenStack 有许多子项目，用于对云计算平台中的各种资源(如计算能力、存储、网络)提供敏捷管理。虽然 OpenStack 刚刚起步，但它也提供了对虚拟化技术的支持。

目前，主流的 Linux 操作系统，包括 Fedora、SUSE 等都支持 OpenStack。OpenStack 在大规模部署公有云时，在可扩展性上有优势，而且也可用于私有云。随着 Ubuntu 12.04 LTS 正式全面使用 OpenStack，OpenStack 将成为基础云平台的第一选择。

1.5 OpenStack 概述

OpenStack 是一款提供公有云和私有云服务的开源平台，它最初是由 Rackspace 和美国国家航空航天局(NASA)共同开发，帮助服务商和企业内部实现类似于 Amazon EC2 的云基础架构服务。前者提供了“云文件”平台代码，该平台增强了 OpenStack 对象存储部分的功能；而后者带来了“Nebula”平台，形成了 OpenStack 其余的部分。另外，Rackspace 还开发了 OpenStack Object Storage，并于 2010 年 7 月将其贡献给 OpenStack，作为其开源子项目，工程代号为 Swift。OpenStack Object Storage(Swift)是开源的，用来创建可扩展的、冗余的对象存储引擎。Swift 使用标准化的服务器存储 PB 级可用数据，但它并不是文件系统(file system)，Swift 看起来更像是一个长期的存储系统(long term storage system)，其目的是获得、调用、更新一些静态的永久性数据，因此 Swift 看起来具有更强的扩展性、冗余性和持久性。美国航空航天局(NASA)的 Ames 研究中心开发了被称作 Nova 的 OpenStack

的雏形，其被开发的目的是为美国的航空航天机构提供可塑性较高的云客户端，之后 Rackspace 涉足了该技术商业化的进程，并最终使其成为一款独立的基础软件。因此，OpenStack 最初是由存储(Swift)和计算(Nova)构成的，如图 1-1 所示。



图 1-1 OpenStack 的组成

Open 为开放之意，Stack 则是堆砌，OpenStack 合起来如其名，就是许多 Open 的 Softwares 堆积的集合，因而该集合系统的功能更为强大。OpenStack 是开源免费的，它由名为 OpenStack Community 的社区开发和维护，这一社区拥有超过 130 家企业及 1350 位开发者，这些机构与个人都将 OpenStack 作为基础设施。OpenStack 社区的首要任务是简化云的部署过程并为其带来良好的可扩展性，采用它来管理云平台中的资源，从而大大降低实施成本。

OpenStack 是一整套开源软件项目的综合体，它允许企业或服务提供者建立、运行自己的云计算和存储设施。Rackspace 与 NASA 是 OpenStack 最初也是重要的两个贡献者，而今，OpenStack 基金会已经有 150 多个会员，包括很多知名公司，如 Canonical、DELL、Citrix 等。

OpenStack 主要用 Python 编程语言编写，整合了 Tornado 网页服务器、Nebula 运算平台，使用 Twisted 软件框架。OpenStack 遵循 Open Virtualization Format、AMQP、SQLAlchemy 等标准。虚拟机软件资源包括 KVM、Xen、VirtualBox、QEMU、LXC 等。最新版本的 OpenStack 代码和文档可以从 OpenStack 官网获取，网址为 <http://www.openstack.org/>。

1.6 OpenStack 的功能与作用

在云计算市场，大多数人都会认同亚马逊云计算服务(AWS)是基础设施即服务(IaaS)的市场领导者，但微软、谷歌和 Joyent 等公司正在试图超越 AWS。这些企业的产品都是闭源的，而 OpenStack 相当于亚马逊 AWS 的开源实现，它承诺最终给用户和服务供应商提供一个大家都可以使用的开源云计算平台，其功能已经包含了 AWS 的核心组件。OpenStack 这样既支持用户的内部私有云，又支持服务供应商的公有云，已初步形成一个生态系统，客户可以自由地在其公有云和私有云之间以及多个供应商之间移动其应用程序和工作负载。

OpenStack 是由不同的功能组件所构成的开源云软件，目前共有 7 个功能不同的组件，分别是运算组件 Nova、对象存储组件 Swift、区块储存组件 Cinder、网络组件 Quantum、身份认证组件 keystone、镜像组件 Glance、前端界面组件 Horizon。就云服务的架构来看，其中又以运算组件、网络组件以及对象存储组件最为重要。

云端服务企业并非必须使用每个组件，根据需要挑选现阶段没有的管理组件即可，例如全球第八大在线零售商 MercadoLibre 使用了运算组件 Nova、对象储存组件 Swift、镜像文件管理组件 Glance 来打造自己的 PaaS 私有云。

OpenStack 每个版本都有不同的名称。在 2010 年 10 月, OpenStack 第一版诞生, 名为 Austin, 而在这个版本中, 仅有运算组件 Nova 与对象储存组件 Swift。随着 OpenStack 的发展以及各领域厂商的加入, 不同的组件推陈出新, 陆续被加入到新的版本内。目前最新版名为 Grizzly, 包含了上述七个组件, 还有两个组件正处于开发阶段, 分别是 Orchestration 层整合组件 Heat 和数据监控组件 Ceilometer。OpenStack 七大组件的功能介绍如表 1-1 所述。

表 1-1 OpenStack 七大组件功能介绍

组件名称	套件功能	Amazon AWS 相似的服务
运算组件 Nova	部署与管理虚拟机	EC2
对象存储组件 Swift	可扩展的分布式存储平台, 以防止单点故障的情况产生, 可存放非结构化的数据	S3
区块存储组件 Cinder	整合了运算套件, 可让 IT 人员查看存储设备的容量使用状态, 具有快照功能	EBS
网络组件 Quantum	可扩展、随插即用, 通过 API 来管理网络架构系统, 以确保 IT 人员在部署云端服务时, 网络服务不会出现瓶颈, 避免影响整个云平台整体性能的发挥	VPC
身份认证组件 keystone	具有中央目录, 能查看哪位使用者可存取哪些服务, 并且提供了多种验证方式	None
镜像组件 Glance	提供硬盘或服务器的镜像文件寻找、注册以及服务交付等功能	VM Import/Export
前端界面组件 Horizon	图形化的网页接口, 让 IT 人员可以观测云端服务目前的规模与状态, 并能够统一存取、部署与管理所有云端服务所使用到的资源	Console

OpenStack 采用 IaaS 服务模式, 让任何人都可以自行建立和提供云端运算服务。此外, OpenStack 也用作建立防火墙内的“私有云”(Private Cloud), 提供给机构或企业内各部门共享资源。

另外, OpenStack 允许在一个单一的平面网络上进行部署, 但是这样并不安全。一般建议使用至少两个网络来进行 OpenStack 的网络部署: 一个用来管理流量, 一个用来进行虚拟机之间的对话。这意味着所有的云计算节点中需要两个网卡(一个运行实例)和网络管理者, 它们运行在不同的 IP 范围中。从虚拟化角度来讲, OpenStack 几乎支持所有的虚拟化程序, 不论是开源的(Xen 与 KVM)还是厂商的(Hyper-V 与 VMware)。但在以前, OpenStack 是基于 KVM 开发的, KVM 常常成为默认的虚拟机管理程序, 两者都使用相同的开源理念与开发方法。

如今, 多数企业用户在 IT 环境中使用了多种虚拟化软件, 有一半的用户选择将开源产品作为性价比更高的虚拟化替代方案。IDC(国际数据公司)报道中指出, OpenStack 是 KVM 增长的一个巨大机会, 同时它具有巨大的行业发展动力, 并拥有一个充满活力的云计算平台社区。有 95% 的 OpenStack 平台由 KVM 驱动, 因此, 随着 OpenStack 的增长, KVM 也会相应增长。

1.7 OpenStack 部署方式介绍

各发行版和 OpenStack 版本的实际部署安装差别很大。一般来说, OpenStack 依赖于一种 64 位 x86 架构,所以具有极低的系统要求。整套 OpenStack 项目可以部署在一个配有 8GB RAM 的单个系统上,但对于一般的 OpenStack 集群,官方建议其至少包含网络节点、控制节点和计算节点,每个节点的配置应该有 12GB RAM、两个 2TB 硬盘和一个网络适配器。其中计算节点(运行虚拟实例)的负载将会有更大的差异,但对于简单的系统而言,一个四核 CPU、32GB 的 RAM 和 2GB 的网络适配器已基本可以满足部署的需求。

此外,部署 OpenStack 云环境大致可分为三类方式:基于虚拟机管理平台的安装方式、基于脚本的安装方式和手动逐步安装方式。表 1-2 所示介绍了安装部署的几种工具。

表 1-2 OpenStack 安装部署工具介绍

工具名称	工具说明	备注
Fuel	Mirantis 出品的部署安装工具,2013 年 10 月份推出 3.2 版本,令人震撼,基本把 OpenStack 所有的部署 Web 化,特别是在网络和存储等方面,用户可以拥有更多的选择	http://www.mirantis.com/
Devstack	Openstack 最早的安装脚本,可以直接通过 git 源码,进行安装,其目的是让开发者可以快速搭建一个云计算环境。目前这套脚本可以在 Ubuntu 和 Fedora 下正常运行	http://docs.openstack.org/developer/devstack/
Stackops	通常在 Linux 下手工安装 OpenStack 比较麻烦,而 Stackops 是一个基于 Ubuntu Linux 操作系统的快速安装 OpenStack 的解决方案。它集成了 OpenStack 的组件,使得 OpenStack 易于安装和设置。安装过程就相当于一个浓缩了的 Ubuntu,只需要选择键盘布局、分区设置 IP 地址等信息	http://docs.stackops.org/display/STACKOPSPRODUCTS/StackOps+Software+Products
Crowbar	Crowbar 是一个 iso 文件,安装完之后可以在 Web 界面进行 OpenStack 的部署,底层使用了 Chef,集成了 Nagios 监控	https://github.com/dellcloud/crowbar
Maas + Juju	Canonical 推出的部署工具,可以用在 OpenStack 的部署上,类似 Puppet、Chef 的部署工具。Maas 是用于安装 Ubuntu 的, Juju 是用于部署应用的	http://www.ubuntu.org.cn/cloud/tools/maas
Rackspace Private Cloud	Rackspace 推出的 OpenStack 部署工具,是一个 iso 文件,里面带一个 Chef 虚拟机。用 iso 安装操作系统的时候就要选择机器的角色。该工具提供了一个诊断工具,让用户可以通过这个诊断工具向 Rackspace 提交问题	http://www.rackspace.com/knowledge_center/product-page/rackspace-private-cloud
Puppetlab	开源的数据中心自动化及配置管理框架,可为系统管理员提供一个易用的平台,进行透明、灵活的系统管理。有了这个平台,系统管理员进行虚拟化和云设施的安装、配置、管理将会变得更加容易	https://github.com/puppetlabs/puppetlabs-openstack
dodai-deploy	OpenStack 部署工具,也是基于 Cobbler 和 Puppet 的,另外还做了一个 Web 管理	https://github.com/nii-cloud/dodai-deploy