

一种基于 OpenStack 的云管理平台

孙 磊, 沈苏彬

(南京邮电大学 计算机学院, 江苏 南京 210003)

摘 要: 当今, 云计算已经成为计算机应用研究领域的一大热点。OpenStack 是当前具有代表性的开源云计算平台之一, 但现有的 OpenStack 管理技术依然存在着以下不足: 对于平台中的各种计算资源缺乏相应的监测功能; OpenStack 集群的扩展依赖人工的部署配置, 易出错且效率低。为解决和优化 OpenStack 平台在管理技术方面的不足, 文中在 OpenStack 开源平台的基础上, 利用 Java Web、自动化脚本控制、异步通信等开发技术, 设计了一套云管理平台。结合 OpenStack 本身提供的丰富 API, 可以实现对平台资源基本管理操作, 同时针对集群扩展和资源监测功能的不足, 设计了相应的开发解决方案。实验结果表明, 对 OpenStack 平台管理方面的优化具有一定效果, 可以有效提升 OpenStack 云平台的管理效率。

关键词: OpenStack; 云管理; 监控; 集群扩展

中图分类号: TP31

文献标识码: A

文章编号: 1673-629X(2016)01-0185-05

doi: 10.3969/j.issn.1673-629X.2016.01.040

A Cloud Management Platform Based on OpenStack

SUN Lei, SHEN Su-bin

(College of Computer, Nanjing University of Posts and Telecommunications,
Nanjing 210003, China)

Abstract: Cloud computing management technology is an important direction in the research of cloud computing in recent years, represented by OpenStack, one of open source cloud computing platforms, has achieved rapid development. But at present it still exists some shortages in cloud management technology of OpenStack, for all kinds of computing resources in the platform, it lacks corresponding monitoring management; the expansion of OpenStack clusters rely on artificial deployment configuration with error-prone and low efficiency. In this paper, on the basis of OpenStack, a cloud management platform was designed by Java Web, automatic control and asynchronous communication and other development technologies, to solve and optimize the defect for OpenStack in management technology. Combined with kinds of API provided by OpenStack, realize the basic management operation for platform resources, at the same time, aiming at the shortcomings of the cluster expansion and resource monitoring function, design the corresponding development solutions. Experimental results show that there are some effects on the optimization of the platform management, the management efficiency for OpenStack cloud platform can be improved effectively.

Key words: OpenStack; cloud management; monitor; cluster expansion

0 引 言

随着当今信息技术的高速发展, 云计算已经成为一个重要热点和趋势。云计算基于虚拟化技术, 由大量物理计算机以及其他硬件组成计算资源池, 提供给用户按需的计算能力、存储能力、网络服务等^[1]。

当今开源 IaaS 云计算平台有 OpenStack, Abi-Cloud, Nimbus, Eucalyputs, CloudStack, OpenNebula 等一些解决方案^[2]。其中 OpenStack 无疑是发展最为迅速的一支, 它由美国国家航天局 NASA 和 Rackspace

共同开发, 作为一款自由开源的软件, 企业和用户都可以使用其构建自身的 IaaS (Infrastructure as a Service) 服务, 具有高扩展性以及部署灵活的特点, 可以实现对企业资源的整合和高效利用^[3]。

IaaS 基础设施即服务层是云计算架构的核心, 它提供给用户对包括 CPU、内存、存储、网络等计算资源的使用, 有效减轻了 IT 运维的成本和复杂性^[4]。云计算架构具有分布式、跨网络、资源种类多的特点, 随之而来为资源管理方面带来了前所未有的挑战, 相比于

收稿日期: 2015-02-27

修回日期: 2015-06-10

网络出版时间: 2016-01-04

基金项目: 江苏省未来网络前瞻性研究资助项目 (BY2013095-1-08)

作者简介: 孙 磊 (1990-), 男, 硕士, 研究方向为云计算应用技术; 沈苏彬, 研究员, 研究方向为计算机应用技术。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20160104.1607.068.html>

传统服务器集群架构,除了对 Web 服务器、应用服务器等物理资源的管理,还需要对 CPU、内存、存储、网络、虚拟机等虚拟资源的统一管理^[5]。监控管理和弹性扩展是云管理平台的重要组成部分^[6],作为一个不断发展成熟的开源项目,OpenStack 在这两部分目前存在一定的缺失和优化的空间。基于 OpenStack 提供的丰富 API 接口,基于 OpenStack 平台进行二次开发,对其相关管理技术进行优化,具有一定的研究意义和实用价值。

1 OpenStack 平台管理技术分析

对资源和服务的有效管理是云计算交付过程中的一个核心要求^[7]。根据云计算架构层次划分可以分为硬件平台管理、虚拟化平台管理、中间件管理、应用管理等,根据功能对象的不同可以分为用户管理、存储管理、网络管理、计算管理等^[4]。当前 OpenStack 平台包含的管理功能主要包括:

计算管理 (Nova): 提供对虚拟机的生命周期管理;

卷存储管理 (Cinder): 提供块存储服务及管理;

认证管理 (Keystone): 提供统一的授权和身份验证管理服务;

镜像管理 (Glance): 提供虚拟机镜像的上传以及存储管理;

网络管理 (Neutron): 提供对平台网络资源的管理^[8]。

根据云计算的五个特征定义,一个云计算平台应该包含按需服务、广泛接入、资源池管理、弹性扩展、计量服务五个特征^[9],当前 OpenStack 平台在按需服务、广泛接入、资源池管理这三个方面均有比较成熟的实现,然而在弹性扩展和计量服务方面,OpenStack 依然存在着一一定的缺失和待优化的空间,其中包括资源监测和集群扩展两个方面。

资源监测是云计算管理技术中的主要环节,其为云平台提供即时的预警处理,同时也是计费 and 故障恢复等操作的前提。监测对象主要包括物理机和虚拟机的 CPU、内存、存储、网络、磁盘等性能数据以及云平台服务组件启停、异常信息的捕获^[10],基于获取到的性能监测数据,可以引导用户或者管理人员进行相关的资源调配、容错处理、动态迁移等操作,实现动态的负载均衡,弹性计算和资源高效利用。OpenStack 平台并没有提供完整成熟的资源监测模块,从 OpenStack Grizzly 版本开始,加入了监控模块 Ceilometer,但这个模块并不成熟,需要其他插件的支持,还不能投入到应用场景中,需要自己开发资源监测模块。

可伸缩性表示云计算对于资源灵活调配和高效利

用的能力^[11]。在 IaaS 层物理硬件资源不足的情况下,需要对 OpenStack 物理服务器集群进行扩展,通常的方式是利用物理服务器部署 OpenStack 相关服务并加入到 OpenStack 服务器集群中,实现 IaaS 层硬件资源池的扩充。OpenStack 节点的部署,涉及多种软件安装和配置调试,较为繁琐复杂,人工部署低效而且易出错^[12]。针对在人工方式下 OpenStack 集群扩展的低效和易错性,以 OpenStack 节点的自动部署功能为核心,为 OpenStack 添加自动化的集群扩展功能模块,可以提高云平台弹性扩展的效率^[10]。

2 方案设计

云管理平台为 OpenStack 各个服务组件和多种计算资源提供了统一的管理接口,同时为上层的 PaaS 平台服务提供所需要的计算资源,总体架构见图 1。

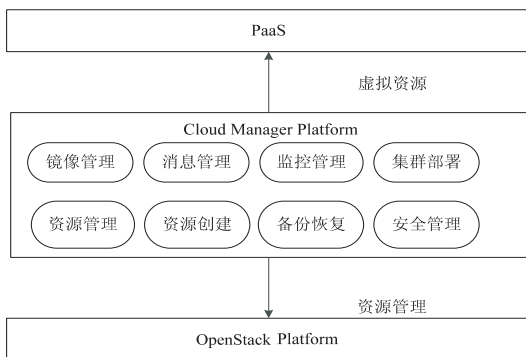


图 1 云管理平台总体架构

云管理平台主要包含以下功能模块:

资源创建: 该模块通过调用 OpenStack 提供的 Rest API 来实现 OpenStack 平台各种计算资源的创建,如虚拟机、网络、存储等。

资源管理: 该模块为 IaaS 层每种资源提供方便、快捷、有效的管理。管理员可以方便地对各种资源进行增删改查等操作。

安全管理: 该模块提供给用户针对 TCP、ICMP、UDP 等多种协议创建多种访问规则。

资源监测: 该模块提供对物理机和虚拟机 CPU、内存、存储、网络四类资源性能指标的检测,同时对 OpenStack 的各个节点上服务组件运行状况进行监测。

消息管理: IaaS 层的运行消息被记录在云平台的消息管理模块中,对各种虚拟资源的操作进行了反馈,使得管理员可以方便地进行问题追踪。

集群扩展: 集群扩展模块提供自动部署 OpenStack 节点,实现 OpenStack 集群扩展的功能。管理员只需配置所要添加物理机的相关信息。

针对 OpenStack 集群扩展功能的设计方案如下: 云管理平台接收添加节点的请求后,建立与目标节点之间的连接,然后将位于控制节点的节点部署所需要

的软件包下载到目标节点并解压,在从控制节点下载执行部署所需要的一系列自动脚本程序,完成 OpenStack 节点的部署并添加到 OpenStack 集群中。集群扩展功能的流程图见图 2。

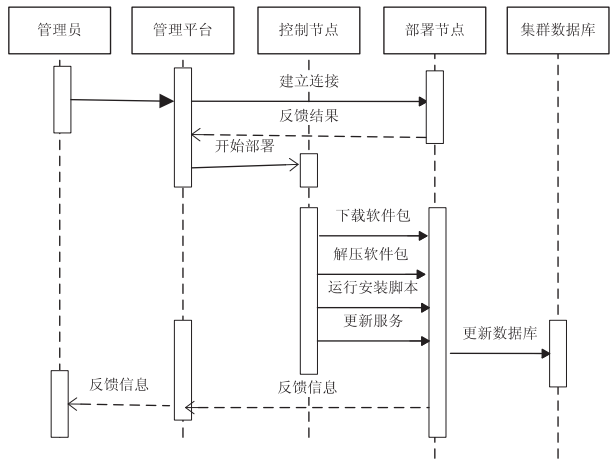


图 2 集群扩展流程

- (1)管理员向管理平台发送添加节点请求。
- (2)根据配置信息中的 IP 地址和 root 账号信息,管理平台向目标主机发送建立连接请求。
- (3)目标主机接收请求,建立连接。
- (4)云管理平台向 OpenStack 控制节点发送创建节点信息。
- (5)目标节点从控制节点下载相关软件包。
- (6)目标节点解压软件包。
- (7)目标节点下载安装脚本,进行自动的安装配置工作。
- (8)目标节点进行自动的服务更新,完成 OpenStack 节点的部署工作。
- (9)部署节点更新集群数据库信息,同时向管理平台发送信息反馈。
- (10)管理平台将添加 OpenStack 集群节点的反馈信息呈现给管理员。

资源监测功能模块负责对 OpenStack 平台中多种计算资源的性能数据进行监测,获取,并将数据信息持久化到监测数据库中。资源监测的功能结构见图 3。

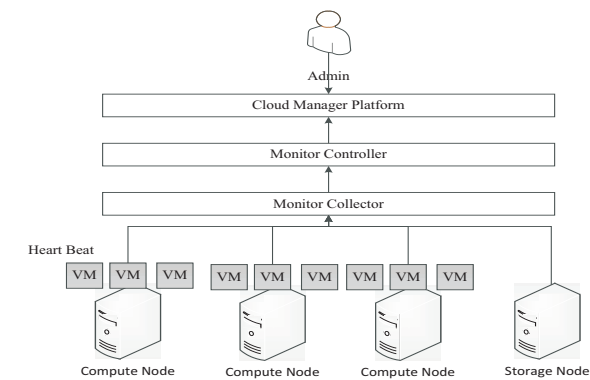


图 3 资源监测功能结构

主要包括:

Monitor Collector: Monitor Collector 是一个守护进程,用来对虚拟机、物理机、服务的状态信息进行采集,并发送给 Monitor Controller,监测信息管理模块。

Heartbeat 机制:心跳机制用来定时获取监测的性能数据信息。Monitor Collector 每隔一段时间对物理机、虚拟机和服务进行数据采集。

Monitor Controller: Monitor Controller 对 Monitor Collector 收集到的资源性能数据信息进行相应管理,一方面将数据信息发送给云管理平台提供实时监测,一方面将监测信息数据持久化到监测数据库中。

3 系统实现

云管理平台基于 Java 语言开发,使用 JavaScript 开发框架 Dojo 以及 j2ssh 等开发包,基于 OpenStack 提供的 RestAPI 实现基本操作功能,然后编程实现资源监测和集群扩展两个功能模块,下面介绍这两个功能模块的实现。

基于上一节中的集群扩展设计方案,下面介绍集群扩展功能的实现算法,流程如图 4 所示。

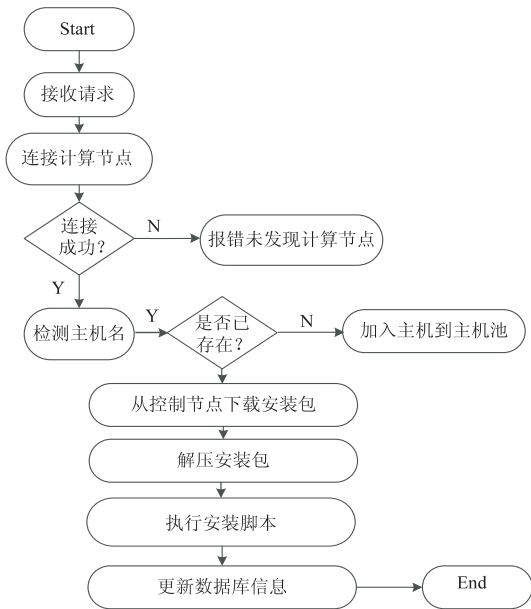


图 4 集群扩展算法流程图

- (1)在云管理平台提交添加节点的请求表单,所需的数据主要包括节点的 IP 地址、用户名和密码。
- (2)根据提交的 IP 地址和管理账户资源,尝试与目标主机进行连接,连接失败提示未发现节点的信息。
- (3)检查目标主机名是否已存在主机池中,即已经包含在 OpenStack 集群中,已包含则提示已包含的信息,否则将主机加入到主机池,继续添加工作。
- (4)目标节点获取控制节点的相关信息,建立与控制节点连接,从控制节点下载软件、相关依赖包以及自动安装脚本到本地。

(5)目标节点解压安装包,并执行自动部署脚本完成安装和部署工作。

(6)完成节点的自动部署,将目标节点添加到 OpenStack 集群中,并更新数据库的信息。

资源监测的对象主要包括物理机、虚拟机和 OpenStack 服务组件,下面对这三部分的实现进行介绍:

物理机监测的实现是利用 SHELL 脚本采集各种资源的状态信息,主要命令实例如下:

(1)功能命令:获取磁盘速率。
命令描述:LANG=C;export LANG;iostat -d -k 1 1 | grep 'sdb\sdc\sdd\sde\sda\sdb\sdc\sdd\sde\sda\sdb\sdc\sdd\sde' | awk '{print \$1,\$2}'

实现原理:利用 linux 的 iostat 命令获取物理机磁盘的读写速率。

(2)功能命令:获取磁盘使用率。
命令描述:LANG=C;export LANG;df -k | awk '{print "DATA_START",\$3,\$4,\$5,\$6,"DATA_END"}'| grep '/DATA_END'

实现原理:利用 linux 的 df 命令获取物理机磁盘使用率。

(3)功能命令:获取内存使用率。
命令描述:LANG=C;export LANG;total=`cat /proc/meminfo | grep "MemTotal:" | awk '{print \$2}'`;free=`cat /proc/meminfo | grep "MemFree:" | awk '{print \$2}'`;echo \$total \$free|awk '{printf("DATA_START %0.2f DATA_END",((100-100*\$2/\$1)))}'`

实现原理:获取 linux 文件系统下/proc/meminfo 中保存的与内存相关的信息,经过计算得出内存使用率。

(4)功能命令:获取 CPU 使用量。
命令描述:LANG=C;export LANG;vmstat | tail -1 | awk '{print "DATA_START",(100-\$15),"DATA_END"}'`

实现原理:利用 linux 的 vmstat 命令获取物理机 CPU 使用量。

虚拟机监测的实现是利用 Libvirt 虚拟化管理库,建立与虚拟机的连接,然后调用 API 来获取虚拟机各种性能状态数据信息,主要的代码实现过程如下:

(1)获取和虚拟机的连接。
conn=new Connect("qemu+tcp://"+hostIp+"/system");
(2)根据虚拟机 Id,探测各个虚拟机的详细信息。
for (int activeDomId : conn.listDomains()) {hq=new HostQuota();Domain domain=conn.domainLookup-

ByID(activeDomId);
(3)调用 Libvirt 相应的 API,获取各种性能数据状态域信息。

获取内存状态信息域:
MemoryStatistic[] ms=domain.memoryStats(5);
获取网络状态信息域:
DomainInterfaceStats dis = domain.interfaceStats(net);

(4)计算资源利用率,根据获取到的性能数据信息,可以计算相应资源利用率。以 CPU 利用率为例,原理是利用 api 返回 CPU 时间,在极短时间内,获取两次 CPU 时间,根据差值计算 CPU 使用率。

服务组件监测的实现是通过调用 OpenStack 相关的 API 获得相关服务组件信息,监测 OpenStack 集群中各节点上服务组件的运行状态,当服务停止或出错时判定该节点从 OpenStack 集群中断线。

4 测试与分析

4.1 测试环境

(1)构建 OpenStack 云平台,测试环境基于 Vmware10 虚拟化软件,由三个 Vmware 虚拟机组成的 OpenStack 集群,分别作为控制节点、计算节点和存储节点。OpenStack 节点配置见表 1。

表 1 OpenStack 平台概况

	控制节点	计算节点	存储节点
CPU 数量	2	4	1
内存/G	2	4	2
硬盘空间/G	20	30	40
网络配置	双网卡	双网卡	双网卡
操作系统	Ubuntu 12.04LTS	Ubuntu 12.04LTS	Ubuntu 12.04LTS

(2)部署云管理平台,云管理平台部署在 OpenStack 计算节点,首先安装 JDK 和 Tomcat 服务器,然后将云管理平台软件包解压到应用目录下。

4.2 功能测试和结果分析

首先对云管理平台的集群扩展功能进行测试,以 IP 地址为 192.168.1.202 的主机为测试对象,在云管理平台中选择自动添加主机功能,主机添加完毕后,在 OpenStack 的控制节点使用 nova-manage service list 指令对当前 OpenStack 集群进行检测,其作用是列出当前 OpenStack 集群中所有节点上 Nova 相关服务组件的运行情况,可以观察到目标主机上的 Nova 服务模块正常启动,目标主机已经加入到 OpenStack 集群中,可以验证集群扩展功能的可行性。

然后对资源监测功能进行测试,登录到资源监测

界面,选择测试环境中的计算节点作为测试对象,获取监测信息数据,成功获取到 CPU 使用率、内存使用率、网口接收发送速率、磁盘读写速率的性能变化曲线。

最后选取七种典型的云计算管理功能,同时对云管理平台和其他两种 OpenStack 管理平台:OpenStack-Horizon 和 Mirantis-Fuel 进行整体功能测试。测试结果对比见表 2,在表中“√”表示该平台对此项功能成功实现,“×”表示未实现。

表 2 云管理平台整体功能测试

	云管理平台	Fuel	Horizon
物理机管理	√	√	×
虚拟机管理	√	×	√
网络管理	√	×	√
云存储管理	√	×	√
物理机监测	√	√	×
虚拟机监测	√	×	×
集群扩展	√	√	×

通过表 2 可以发现,云管理平台对七种管理功能均完成了实现,OpenStack-Horizon 缺乏资源监测和集群扩展功能;Mirantis-Fuel 缺乏对虚拟资源的管理功能。

综上所述,云管理平台为 OpenStack 平台提供了比较成熟的管理框架,并实现了一定的管理方面的优化,具有一定的应用价值和实践意义。

5 结束语

文中通过对云计算管理和开发技术的深入研究和分析,设计了基于 OpenStack 的云管理平台。在 OpenStack 平台基本管理操作的基础上,设计开发了资源监测和集群扩展功能模块。通过相关测试,验证了云管理平台在资源管理、资源监测和集群扩展功能方面具有一定的先进性和应用价值。未来,可以对云平台的高可用性管理、容错处理等进行进一步的优化和研究。

参考文献:

[1] Armbrust M, Fox A, Griffith R, et al. A view of cloud compu-

ting[J]. Communications of the ACM,2010,53(4):50-58.

[2] Rimal B P, Choi E, Lumb I. A taxonomy and survey of cloud computing systems[C]//Proc of fifth international joint conference on INC, IMS and IDC. [s. l.]:[s. n.],2009:44-51.

[3] Feller E, Rilling L, Morin C. Snooze: a scalable and autonomic virtual machine management framework for private clouds [C]//Proceedings of the 2012 12th IEEE/ACM international symposium on cluster, cloud and grid computing. [s. l.]: IEEE Computer Society,2012:482-489.

[4] Sotomayor B, Montero R S, Llorente I M, et al. Virtual infrastructure management in private and hybrid clouds[J]. IEEE Internet Computing,2009,13(5):14-22.

[5] VMware Inc. VMware vSphere, the first cloud operating system, provides an evolutionary, non-disruptive path to cloud computing[M]. [s. l.]: VMware Inc,2009.

[6] Bellavista P, Giannelli C, Mattetti M. A practical approach to easily monitoring and managing IaaS environments[C]//Proc of IEEE symposium on computers and communications. [s. l.]: IEEE,2013.

[7] Wibowo E. Cloud management and automation[C]//Proc of joint international conference on rural information & communication technology and electric-vehicle technology. [s. l.]: IEEE,2013:1-4.

[8] Liu X, Qiu X, Xie X, et al. Implement of a light-weight integrated virtualized environment manager for private cloud computing[C]//Proc of international conference on computer science & service system. [s. l.]: IEEE,2012:523-526.

[9] Zhang Z, Zhou W, Qian L, et al. CloudMaster: a cloud computing management system of BigCloud[C]//Proc of OCS. [s. l.]: IEEE,2012:40-42.

[10] Clay R W, Wild N R, Bird D J, et al. A cloud monitoring system for remote sites[J]. Publications of the Astronomical Society of Australia,1998,15(3):332-335.

[11] Forell T, Milojicic D, Talwar V. Cloud management: challenges and opportunities[C]//Proc of IEEE international symposium on parallel and distributed processing workshops. [s. l.]: IEEE,2011:881-889.

[12] Morariu O, Borangiu T, Morariu C. Multi-layer QoS monitoring in private clouds[C]//Proc of 24th international workshop on database and expert systems applications. [s. l.]: IEEE,2013:236-240.

一种基于 OpenStack 的云管理平台



作者：[孙磊](#)，[沈苏彬](#)，[SUN Lei](#)，[SHEN Su-bin](#)
作者单位：[南京邮电大学 计算机学院, 江苏 南京, 210003](#)
刊名：[计算机技术与发展](#)[ISTIC](#)
英文刊名：
年，卷(期)：2016(1)

引用本文格式：[孙磊](#).[沈苏彬](#).[SUN Lei](#).[SHEN Su-bin](#) 一种基于 OpenStack 的云管理平台[期刊论文]-[计算机技术与发展](#) 2016(1)