



私有云平台服务能力检测方法

柳春懿*, 张 晓, 李阿妮, 陈 震

(西北工业大学 计算机学院, 西安 710129)

(*通信作者电子邮箱 18729580703@163.com)

摘 要: 针对现阶段使用私有云评测方法缺失所导致搭建私有云供需不匹配的问题, 提出了一种自适应、可扩展的私有云系统能力检测方法, 可对基础设施即服务(IaaS)云计算系统能力进行评测。首先, 通过私有云应用程序接口动态扩展虚拟机数量; 然后, 通过性能特征模型选取虚拟机所需配置的硬件信息和操作系统类别, 并根据用户的需求差异使用不同的负载模型, 形成相应的模拟运行环境; 最后, 使用云计算服务协议(SLA)作为测试标准来衡量私有云服务能力。使用所提方法对 Openstack 开源私有云平台的能力进行了评测, 实验结果表明, 所提方法可以准确得出私有云服务能力, 结果和用户实测一致。该方法负载生成成本更低, 测试效率更高。并将该方法与 Openstack 自带的组件 Rally 进行对比, 拓展性和负载动态模拟有很大的改进。

关键词: 私有云; 服务能力; 基础设施即服务; Openstack 云平台; 云测试

中图分类号: TP301 **文献标志码:** A

Service capacity testing method of private cloud platform

LIU Chunyi*, ZHANG Xiao, LI Ani, CHEN Zhen

(School of Computer Science, Northwestern Polytechnical University, Xi'an Shaanxi 710129, China)

Abstract: Concerning the problem that the lack of testing methods would lead to mismatch between supply and demand of private clouds, an adaptive and scalable private cloud system testing method was proposed, which can test private cloud computing ability in IaaS (Infrastructure as a Service). The number of virtual machines was dynamically increased through the private cloud application program interface, hardware information and operating system category of the virtual machine configuration were selected by performance-characteristic model, and different load models were used according to different needs of users to form simulation environment. At last, cloud computing Service Level Agreement (SLA) was used as a test standard to measure the ability of private cloud services. The proposed method was implemented in Openstack. The experimental results show that private cloud platform service capacity can be obtained by the proposed method with lower cost and higher efficiency than user test. Compared with Openstack component Rally, scalability and dynamic load simulation of the proposed has greatly been improved.

Key words: private cloud; service ability; Infrastructure as a Service (IaaS); Openstack cloud platform; cloud testing

0 引言

随着云计算^[1]的不断发展, 云生态圈越来越完善。许多企业因云资源利用高效和部署灵活等特点, 使用云服务方式部署应用。根据云计算服务性质的不同, 大部分云计算可以分为公有云和私有云。公有云虽然使用方便, 但在安全性上的不足, 使得私有云占领云计算的绝大部分市场。VMware 公司便是现阶段市场的领头羊。对比开源私有云, 该平台使用价格昂贵, 开放资源较少。因此开源私有云市场份额^[1]大有迎头赶上趋势, Openstack、Eucalyptus、Cloudstack 等开源私有云社区备受企业和开发者的关注。

针对私有云使用者而言, 最为关注的是私有云平台服务能力能否满足业务需求。他们希望在满足企业基本需求的情况下, 尽可能降低维护购买成本, 因此, 根据用户实际需求准

确地检测出私有云系统的服务能力显得尤为重要^[2]。

在云环境下, 用户使用场景种类繁多导致机器配置灵活。传统的服务器服务能力检测方法应用于私有云平台检测面临如下问题: 1) 测试难度加大。私有云服务能力检测是一个耗时耗力的繁重专家任务, 没有私有云相关的基础技术和多年运维经验, 不能准确地根据复杂的使用情况得出相应的结论。2) 测试结果不准确。对每一个虚拟机进行不同处理, 虚拟机之间会互相争夺计算机硬件资源。3) 测试通用性较差。目前, 一些大型云服务商已经拥有完整的平台服务能力检测, 但是由于各个服务商的云平台实现不同, 他们的方法只能应用于自己平台, 不适用于其他云平台, 缺乏统一性。目前, 一些大型云计算服务商已经拥有自己专门的一套云需求检测手段来检测最大用户支持度, 但是由于不同公司的云平台实现不同, 因此缺乏统一性; 而对于很多没有云服务能力检测的计算

收稿日期: 2016-07-15; **修回日期:** 2016-11-21。 **基金项目:** 国家 863 计划项目(2015AA015307); 国家自然科学基金资助项目(61472323); 中央高校基本科研业务费专项(3102015JSJ0009)。

作者简介: 柳春懿(1993—), 男, 安徽滁州人, 硕士研究生, 主要研究方向: 云计算、云存储; 张晓(1978—), 男, 河南新乡人, 副教授, 博士, CCF 会员, 主要研究方向: 云计算、云存储、海量存储; 李阿妮(1992—), 女, 陕西咸阳人, 硕士研究生, 主要研究方向: 云计算、云存储; 陈震(1992—), 男, 安徽安庆人, 硕士研究生, 主要研究方向: 云存储。



机相关公司,则存在云服务质量无法保证的问题。对用户而言,需要及时获取云服务器的最大支持度。对商家来说,提供充足云组件的性能信息是非常必要的,并且建立高质量云计算服务很明显已经成为一个重要的研究领域^[3]。

针对以上问题,本文提出了一种私有云平台服务能力快速检测模型,可以根据不同的用户群体和需求,自动部署测试程序,准确快速地获取私有云服务能力。本文的主要工作体现在:1)提出了完整的私有云平台服务能力快速检测方法 FCtest。通过基础设施即服务 (Infrastructure as a Service, IaaS) 的应用程序接口 (Application Program Interface, API) 和交互式命令 expect,根据用户需求自动部署并测试,节约了测试总体时间并简化了测试用户的操作。2)提出了基于负反馈机制的多线程负载生成模型。只针对内存和 CPU 进行负载生成,达到快速负载生成。通过对 CPU 的动态控制,减少其他操作对 CPU 的波动,从而保证了测试结果的准确性。3)提出了场景应用模型和评估模型。通过抽象模型应用到不同用户使用环境中,使得该服务能力检测方法适用于绝大多数私有云平台。总之,本文提供一种客观、公正、高效、通用的面向第三方的私有云平台服务检测方法,来简化检测人员操作,提高检测结果的准确性,适用不同私有云平台。解决此问题不仅能够增强私有云用户和管理的使用信心,减少实际的资源浪费,更进一步推动了私有云模式的完善。

1 相关工作

文献[4]中提出了云计算服务等级协议,并针对 IaaS 介绍了相关的服务等级协议 (Service Level Agreement, SLA) 参数,主要涉及到虚拟机和服务器以及用户需求响应时间相关的参数,但是没有具体描述参数如何筛选获取。在此基础上,张健^[5]提出了针对服务的 SLA,如存储即服务的 SLA 等。为了降低云测试的难度,不少研究人员从管理者角度发表了自己的一些评测研究^[6-10],文献[11-12]实现了一个称为 SQL 性能分析器 (SQL Performance Analyzer, SPA) 的工具,用于自动测量分析存储的性能,能够采集的指标有 IO 吞吐率和响应时间,使用 Benchmark 控制器组件,通过 SSH (Secure Shell) 连接虚拟机,然后执行目标工作负载,最后将收集到的结果存入 SQLite 数据库中。它针对的只是存储系统,需要在目标机器上执行定制的 Benchmark 才能获得性能指标参数,并且工具的安装与配置可能比较复杂,时间成本较大。陈斌等^[13]在他的云环境下基于代理服务的性能收集模型中,提出了一种评价云中心的性能指标分析架构,利用客户端、代理端、服务端获取性能并最终通过它们获得云中心的性能情况;但是,该模型基于代理服务接口,会造成额外的工作量和成本,并且通过 3 个不同的性能指标来计算得出私有云的需求,结果不准确。文献[14]中提出性能测试的两个层面:性能预测和性能实时状态检测。文献[15]提出云计算的性能分析模型可以被描述为一个近似分析模型,该模型建立在阻塞预警控制和完全拒绝策略为保证的前提下。文献[16]提出了功能、性能、压力三方面测试用例的设计方法。文献[17]提出了创新云服务模式“测试即服务”(Testing as a Service, TaaS)。

私有云提供商 Openstack 社区的 Tempes 项目为其私有云提供了功能测试和集成测试,主要是基于 unittest 和 nose 来实

现,通过调用 Openstack API 来测试功能,响应验证。其中该项目提供压力测试,主要测试 Openstack 在高工作负载中运行时出现什么问题。针对性能,Openstack 开发出 Rally 组件。该方法确实可以准确检测出最大用户使用数量,但是不能尝试不同的负载情况下的支持度,测试范围有限。

聂规划等^[18]从客户的角度出发,以客户的云需求波动为背景引入一种新的实例服务模式,提出了一种新的组合决策模型。该模型主要是从商业的角度出发,没有考虑到私有云平台的性能细节,只是单单从成本的变化趋势进行分析。苏晶^[19]也提出了针对私有云 Openstack 的系统性能需求,支持尽可能多的用户的方案,根据企业用户数需求、网络带宽决定,至少同时能容纳 200 个用户,高于 200 之后响应时间会增加,但能处理用户请求;但得出的结论没有严格的证明,所以准确性不能得到保证。

在以上研究的基础上,本文提出一种通用的私有云平台服务能力检测方法,快速准确地解决实际需求信息获取困难的问题。

2 检测模型

用户实际使用情况模拟和能力评估方法是私有云服务能力检测的非常重要的步骤,它有利于更准确地获取相关信息。首先,抽象出服务能力检测模型的整体框架和流程,以适应不同的私有云平台,简化用户操作;然后,提出了无干扰的多线程负载生成和检测环境部署的方法,提供多种部署接口;最后,针对不同的私有云平台,根据用户不同要求提供统一的评估模型。

2.1 检测框架和流程

图 1 给出了私有云环境下服务能力检测逻辑框架,分为 4 个层次。

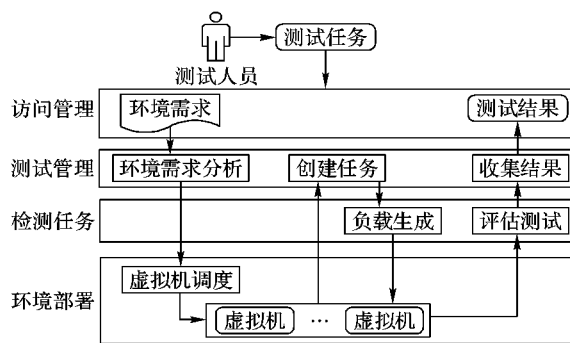


图1 私有云服务能力检测逻辑框架

Fig. 1 Logical framework of private cloud service ability testing

1)访问管理模块。它是云测试服务的使用者和云测试服务进行交互的接口。用户通过简单的输入操作提交测试请求到测试管理模块。用户请求服务时提交测试环境需求,系统返回给用户检测结果。

2)测试管理模块。该模块对用户环境需求进行需求分析,得到具体的 (M, C) 二元向量组。其中 M 是内存相关向量参数, C 是CPU相关向量参数。根据请求申请虚拟机,创建测试任务请求并收集最后结果。

3)检测任务模块。该模块只要负责私有云服务能力检测核心,负载生成和评估测试自动安装并运行。

4)环境部署模块。主要功能提供 IaaS 服务。根据测试



管理模块的需求分析结果分配多台虚拟机模拟用户规模、用户机器配置和使用情况等。

本架构针对的是云服务器的计算需求性能检测,所以针对计算节点进行负载变化,忽略网络、存储、控制对计算节点的影响。其中用户只需要将环境需求输入到系统中,该系统会根据用户的不同选择方式,自动给出环境需求文档;将文档放入下一个模块进行分析,得出模拟计划,根据模拟计划分配虚拟机数量、配置等性能信息;同时在环境部署完成之后,启动测试程序,完成后返回测试结果。综上针对用户而言,底层的分析和部署都是自动的。

2.2 性能特征模型

在向量参数选择上,本文认为应该注意以下几点:所选的参数应该可以直接反映出任务的计算负载情况,并且必须是容易观察和控制;可以通过所选的参数调节虚拟机性能;应该需要确定能够捕捉任务计算性能的最小参数集合。考虑参数之间的互相影响,内存的参数变化可能会导致 CPU 相关参数的变化。

M 性能向量使用 (M_{actual}, F_m) 二维向量。其中 M_{actual} 为实际内存使用量, F_m 为内存波动大小,可以用如下公式表示:

$$M_{\text{actual}} = M_{\text{total}} * U_m$$

其中: M_{total} 为任务所在服务器内存总容量, U_m 为该环境下内存使用率。

C 性能向量使用 $(N_{\text{cpu}}, N_{\text{core}}, F_c, T_c)$ 四维向量描述,其中 N_{cpu} 为 CPU 数量, N_{core} 为每一个 CPU 的核数, F_c 为每一个 CPU 使用率的波动情况, T_c 为 CPU 每个核的每秒运行的指令数, N_{total} 是机器中所有 CPU 总核数。描述每一个任务的 CPU 负载情况,可以用如下公式表示:

$$N_{\text{total}} = N_{\text{core}} * N_{\text{cpu}}$$

$$C_{\text{actual}} = \sum_{c=1}^{N_{\text{total}}} T_c$$

因为 CPU 每个核的每秒运行指令数难以获得,所以本文提出以下公式:

$$C_{\text{actual}} = \sum_{\text{cpu}=1}^{N_{\text{total}}} U_{\text{cpu}} * G_{\text{cpu}}$$

其中: U_{cpu} 是 CPU 使用率; G_{cpu} 为 CPU 每秒运行命令行数,即浮点计算能力。

为简便使用,本文忽略物理机与虚拟机映射差异。虚拟机中内存与 CPU 相关参数计算方式与上文一致。

将以上参数作为输入参数,模拟集群的实际内存和硬件需求作为输出参数,对实际的硬件信息进行模拟、评测。

2.3 场景模拟模型

本文提出了 6 种需求模拟,涵盖不同的典型使用场景,给私有云服务能力检测提供用户友好的部署接口。针对不同的用户给出不同的需求获取如表 1 所示。

表 1 需求模拟表
Tab. 1 Demand simulation

负载模拟	功能
历史情况模拟	利用历史数据采集工具,采集并完成基本负载模拟运行
泊松分布模拟	利用性能负载的泊松分布来模拟整个系统的运行情况
均值分布模拟	利用平均的性能负载分布来模拟整个系统运行情况
用户输入模拟	用户可以根据自己的专业判断,直接输入参数的值
用户选择模拟	用户根据自己使用的常用软件自动推断出参数来模拟
混合模型模拟	不同的虚拟机拥有不同的负载情况

本方案从用户规模、虚拟机硬件、虚拟机负载三个方面模拟真实的用户使用场景。针对用户规模,可以利用 IaaS 私有云提供的 API 进行横向的扩展。参考私有云 Openstack 提供有关虚拟机动作操作,主要包括创建、删除等操作。

针对虚拟机硬件,私有云平台提供丰富的 API,可以灵活修改虚拟机的硬件配置。以 Openstack 的 rest API 为例:通过 POST 模式下/v2.1/{tenant_id}/servers 改变 flavorRef 参数,控制虚拟机类型,即改变内存大小、虚拟核数、硬盘大小等资源,即用户可以定制云主机通过改变 imageRef 参数,改变虚拟机操作系统种类。除此之外,Openstack 提供五种云主机类型,涵盖不同的虚拟机配置类型,如表 2 所示。

表 2 Openstack 云主机类型
Tab. 2 Openstack cloud type

云类型	虚拟内核数	根磁盘/GB	内存/GB	磁盘/GB
tiny	1	1	1	1
small	1	20	2	20
medium	2	40	4	40
large	4	80	8	80
xlarge	8	160	16	160

针对虚拟机负载,多台虚拟机负载的快速生成和稳定运行将直接影响私有云服务能力检测准确性,所以本文提出了一种无干扰的多线程负载生成方法。因为申请和释放内存会影响 CPU 的负载,故采用两个线程分别控制 CPU 和内存负载。CPU 线程采用负反馈技术实时调整 CPU 利用率,降低内存分配对 CPU 负载的影响。两个线程的处理流程和关系如图 2 所示。

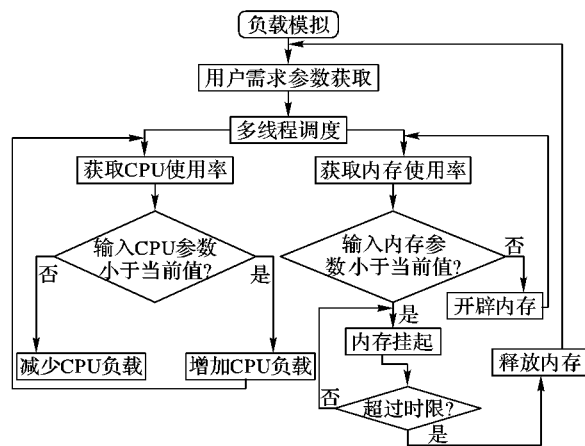


图 2 负载生成模块示意图

Fig. 2 Workload generation module

2.4 测试设计

为了精确地描述私有云服务能力,本文根据云计算服务协议(SLA)作为标准来衡量。其中主要参数包括:

- 1) T_{in} 导入时间。虚拟机正式使用前的准备和导入时间。
- 2) T_{add} 规模扩大时间。增加一台虚拟机数量的时间。
- 3) T_{response} 业务响应时间。业务提供商响应和处理用户请求的时间。
- 4) P 性能参数。单个虚拟机在共存虚拟机下的性能分数。

在实际测试过程中 $T_{\text{in}} \ll T_{\text{add}}$, 并且基于 IaaS 的云计算增



加虚拟机一定会作使用前的准备和导入,所以用 $(T_{add}, T_{response}, P)$ 三维向量来评估私有云服务能力,并评估出该环境下的最大用户支持度。最大用户支持度是指私有云提供虚拟机给用户正常使用的最大数量,用来衡量私有云服务能力。其中 P 性能分数使用 Benchmark 进行测试衡量。

根据用户的实际体验不同,可以采用主动模式和被动模式来获取:主动模式即将测试虚拟机交给用户实际使用,根据用户的主观判断,检测出最大用户支持度;被动模式即使用软件工具来代替用户的主观判断,通过客观数据分析,检测出最大用户支持度。

将同一私有云环境下负载生成后的虚拟机称为负载虚拟机群,运行检测程序的虚拟机称为测试虚拟机。通过动态调整虚拟机的数量,控制虚拟机的增量,得到检测结果的集合。测试过程自动迭加完成,保证了测试的速度,简化了用户的操作。其中 Num 为当前虚拟机数量, $step$ 为每次增加虚拟机数量。具体的测试算法如下:

```
Num = 0
While True:
    测试( $T_{add}, T_{response}, P$ ) 在该环境下值;
    IF  $step == 0$ :
        输出虚拟机的数量  $Num$ ;
        Break;
    IF ( $T_{add}, T_{response}, P$ ) 劣于标准:
        释放  $\lfloor step/2 \rfloor$  台虚拟机资源
         $Num -= \lfloor step/2 \rfloor$ 
         $step = \lfloor step/2 \rfloor$ 
    ELSE:
        创建  $step$  台虚拟机;
         $Num += step$ ;
```

3 实验结果与分析

3.1 测试环境

本次评测在 Openstack 私有云环境下进行,操作系统为 Ubuntu 14.04 LTS,内核版本为 3.13.0-24-generic, KVM 版本为 2.0.0, Openstack 版本为 Juno,评测的配置参数如表3。

表3 测试机器配置
Tab. 3 Tested computer configuration

节点名称	CPU	内存/GB	硬盘/GB
计算节点	80核	160	600(SSD)
控制节点	双路4核	16	150
网络节点	双路4核	16	150
虚拟机配置	单核	1	10

3.2 测试实验方案

根据实验者使用特定的负载波动、操作系统、负载情况等,本次测试选取了3种不同应用情况,将该应用情况分别与负载生成模型相对应。比较在不同的应用情况下,测试私有云计算节点服务能力,如表4所示。

表4 云测试方案
Tab. 4 Cloud testing scheme

方案	模式	针对应用	内存使用率	CPU使用率	类型	操作系统
1	用户选择	办公	无	无	tiny	Ubuntu
2	历史情况	软件开发	无	无	small	Windows
3	均值分布	图形图像	80%	80%	medium	Windows

3.3 实验结果

实验一 本文根据上文所提出的评估模型,采用被动模式。利用 unixbench 工具,以跑出的分数为结果集分数,500分为评判标准分数。对方案1进行实验,得出空载虚拟机数量为232。除此之外,对比3种不同的方案,得出该环境下计算节点方案1的最大用户使用度为180,方案2的情况最大用户使用度为122,方案3的情况下最大用户使用度为85,如图3所示。

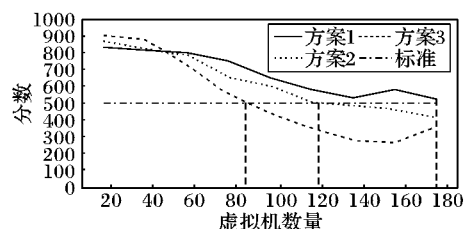


图3 不同场景支持情况比较

Fig. 3 Comparison of different scenarios

实验二 根据实际的机器性能,可以采用被动模式验证。在方案2同样的情况下,研究测试了不同负载条件下的编译时长,使用的编译包为 php-5.6.18。取平均值结果如图4所示。评估模型的结果集分数采用编译该文件的时长。当编译时长为标准时长2倍时,定为其评判标准,得出最大用户使用度为127。

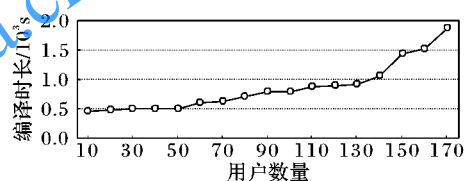


图4 编译时长验证图

Fig. 4 Verification of compiling time

实验三 不改变实验环境情况下,采集方案2中 T_{add} 如图5可知,将检测时间分为那台机器的规模扩大时间和业务响应时间两个时间段。总部署时间不超过30min。部署时间为虚拟机分配和负载生成的时间,通过比较可以发现,负载生成和部署非常快捷。

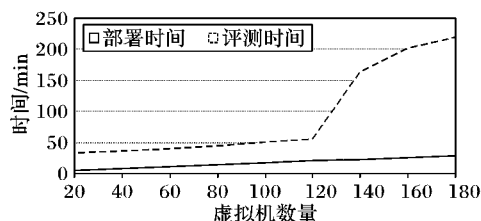


图5 检测耗时统计图

Fig. 5 Cost time statistics

实验四 在同条件下,在用户数为100的情况下,使用 FCtest 和 Rally 对同样的方案2进行测试。其中运行 Rally 的测试运行时间是1423s, FCtest 的部署时间是1271s,测试时间是1730s。除此之外,本文对比了 FCtest 和 Rally 相关功能。

结合实验三、四对比可以发现,FCtest 的部署近似线性,而 Rally 测试会出现较大的波动,并且 FCtest 在一定时间内的评测响应时间与用户数量正相关。若加上评测时间, Rally 的评测响应更佳, Rally 不能添加特定负载进行测试,所以本文



只比较了两个的部署时间。

表5 Raaly 和 FCtest 功能对比表

Tab. 5 Function comparison of Raaly and FCtest

测试方案	测试项目	Rally	FCtest
实际测试	特定数量测试	✓	✓
	并行测试	✓	✓
	响应时间测试	✓	✓
模拟测试	拓展性模拟	×	✓
	操作系统模拟	✓	✓
	硬件模拟	✓	✓
	负载模拟	×	✓

4 结语

本文提出了一种私有云服务能力检测的通用方法。通过 IaaS 私有云平台灵活配置特点,自动搭建测评环境,简化了用户测评操作过程。采用基于负反馈机制的负载生成方法,可以快速获得恒定的负载,提高了检测的准确性。提供了通用的私有云检测模型,改善了传统方法不适应不同私有云平台的缺陷。实验结果表明,该方法在部署和负载生成上提高了效率,节省了评测成本。私有云服务能力检测方法可随规模不同自动部署,并根据用户的需求使用不同的负载模型评测出最大用户使用度和空载虚拟机数量,基本满足用户对私有云服务能力检测的需求。

以后的工作中,将从以下方面来改进测试方法。本文针对的是私有云服务能力中的计算能力,没有对网络的测试。在实际使用情况下,网络对用户使用存在一定的影响,所以网络的有关评测将作为后期研究的重点。

参考文献 (References)

- [1] 吴吉义, 沈千里, 章剑林, 等. 云计算: 从云安全到可信云[J]. 计算机研究与发展, 2011, 48(S1): 229-233. (WU J Y, SHEN Q L, ZHANG J L, et al. Cloud computing: cloud security to trusted cloud[J]. Journal of Computer Research and Development, 2011, 48(S1): 229-233.)
- [2] 姜毅, 王伟军, 曹丽, 等. 基于开源软件的私有云计算平台构建[J]. 电信科学, 2013, 29(1): 68-75. (JIANG Y, WANG W J, CAO L, et al. Construction of private cloud computing platform based on open source software[J]. Telecommunications Science, 2013, 29(1): 68-75.)
- [3] ZHANG Y L, ZHENG Z B, LYU M R. Real-time performance prediction for cloud components[C]// Proceedings of the 2012 IEEE 15th International Symposium on Service-Oriented Real-Time Distributed Computing Workshops. Piscataway, NJ: IEEE, 2012: 106-111.
- [4] ALHAMAD M, DILLON T, CHANG E. Conceptual SLA framework for cloud computing[C]// Proceedings of the 4th IEEE International Conference on Digital Ecosystems and Technologies. Piscataway, NJ: IEEE, 2010: 606-610.
- [5] 张健. 云计算服务协议(SLA)研究[J]. 电信网技术, 2012(2): 7-10. (ZHANG J. Study on cloud computing SLA[J]. Telecommunication Network Technology, 2012(2): 7-10.)
- [6] ANTONIOU A. Performance evaluation of cloud infrastructure using complex workloads[D]. Delft: Delft University of Technology, 2011: 1011-1026.
- [7] HILL Z, LI J, MAO M, et al. Early observations on the performance of windows azure[C]// Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing. New York: ACM, 2010: 367-376.
- [8] DELMERICO J A, BYRNESY N A, BRUNOZ A E, et al. Comparing the performance of clusters, Hadoop, and active disks on microarray correlation computations[C]// Proceedings of the 2009 International Conference on High Performance Computing. Piscataway, NJ: IEEE, 2009: 378-387.
- [9] COOPER B F, SILBERSTEIN A, TAM E, et al. Benchmarking cloud serving systems with YCSB[C]// Proceedings of the 1st ACM Symposium on Cloud Computing. New York: ACM, 2010: 143-154.
- [10] ZHANG X, FENG W, QIN X. Performance evaluation of online backup cloud storage[J]. International Journal of Cloud Applications and Computing, 2013, 3(3): 20-33.
- [11] NOORSHAMS Q, BRUHN D, KOUNEV S, et al. Predictive performance modeling of virtualized storage systems using optimized statistical regression techniques[C]// Proceedings of the 4th ACM/SPEC International Conference on Performance Engineering. New York: ACM, 2013: 283-294.
- [12] BRUHN D. Modeling and experimental analysis of virtualized storage performance using IBM system z as example[D]. Karlsruhe, Germany: Karlsruhe Institute of Technology, 2012.
- [13] 陈斌, 王志坚, 王宇. 云环境下基于代理服务的性能收集模型[J]. 微型机与应用, 2013, 32(16): 57-60. (CHEN B, WANG Z J, WANG Y. Performance collection model with Agent and server interface for cloud computing[J]. Microcomputer & its Applications, 2013, 32(16): 57-60.)
- [14] KHAZAE H. Performance analysis of cloud centers under burst arrivals and total rejection policy[C]// Proceedings of the 2011 IEEE Global Telecommunications Conference. Piscataway, NJ: IEEE, 2011: 1-6.
- [15] RAAJESWARI R, SELVARANI R. A performance analysis method for service-oriented cloud applications (SOCAs)[C]// Proceedings of the 2012 International Conference on Computer Communication and Informatics. Piscataway, NJ: IEEE, 2012: 1-7.
- [16] 冯昔川. 基于 Cloudstack 云平台的 Pfense 软件防火墙的分析与测试[D]. 北京: 北京邮电大学, 2015. (FENG M C. Analysis and testing of the Pfense software firewall based on the Cloudstack cloud platform[D]. Beijing: Beijing University of Posts and Telecommunications, 2015.)
- [17] 张一弛. 云计算环境下的“测试即服务”模式及其实现[D]. 武汉: 华中师范大学, 2013. (ZHANG Y C. Research and implement of testing as a service mode on cloud computing[D]. Wuhan: Central China Normal University, 2013.)
- [18] 聂规划, 余其平, 陈冬林. 客户云需求波动下的多实例组合决策研究[J]. 计算机工程, 2013, 39(5): 42-47. (NIE G H, SHE Q P, CHEN D L. Study on multi-instance combination decision under user cloud demands fluctuation[J]. Computer Engineering, 2013, 39(5): 42-47.)
- [19] 苏晶. 面向 OpenStack 的企业云管理平台的研究与实现[D]. 成都: 电子科技大学, 2014. (SU J. Research and implementation of enterprise cloud management platform for OpenStack[D]. Chengdu: University of Electronic Science and Technology of China, 2014.)



参考文献 (References)

- [1] DULLOOR S R, KUMAR S, KESHAVAMURTHY A, et al. System software for persistent memory[C]// Proceedings of the 9th European Conference on Computer Systems. New York: ACM, 2014: Article No. 15.
- [2] CONDIT J, NIGHTINGALE E B, FROST C, et al. Better I/O through byte-addressable, persistent memory[C]// Proceedings of the 22nd ACM Symposium on Operating Systems Principles. New York: ACM, 2009: 133 – 146.
- [3] FREITAS R F, WILCKE W W. Storage-class memory: the next storage system technology[J]. IBM Journal of Research & Development, 2008, 52(4/5): 439 – 447.
- [4] XU J, SWANSON S. NOVA: a log-structured file system for hybrid volatile/non-volatile main memories[C]// Proceedings of the 14th USENIX Conference on File and Storage Technologies. Berkeley, CA: USENIX Association, 2016: 323 – 338.
- [5] OU J, SHU J, LU Y. A high performance file system for non-volatile main memory[C]// Proceedings of the 11th European Conference on Computer Systems. New York: ACM, 2016: Article No. 12.
- [6] 刘波, 宋志棠, 封松林, 等. 我国相变存储器的研究现状与发展前景[J]. 微纳电子技术, 2007, 44(2): 55 – 61. (LIU B, SONG Z T, FENG S L, et al. Current situation and developing trend on phase-change memory in China[J]. Micronanoelectronic Technology, 2007, 44(2): 55 – 61.)
- [7] SCOTT J F. 铁电存储器[M]. 朱劲松, 吕笑梅, 朱旻, 译. 北京: 清华大学出版社, 2004: 1 – 200. (SCOTT J F. Ferroelectric Memories[M]. ZHU J S, LYU X M, ZHU M, translated. Beijing: Tsinghua University Press, 2004: 1 – 200.)
- [8] 田晓波. 忆阻器电路特性与应用研究[D]. 长沙: 国防科学技术大学, 2009. (TIAN X B. Study on the characteristic and application of memristor circuits[D]. Changsha: National University of Defense Technology, 2009.)
- [9] HUAI Y. Spin-Transfer Torque MRAM (STT-MRAM): challenges and prospects[J]. AAPPS Bulletin, 2008, 18(6): 33 – 40.
- [10] Breakthrough nonvolatile memory technology[EB/OL]. [2016-08-10]. <https://www.micron.com/about/emerging-technologies/3d-xpoint-technology>.
- [11] VOLOS H, TACK A J, SWIFT M M. Mnemosyne: lightweight persistent memory[J]. ACM SIGARCH Computer Architecture News, 2011, 39(1): 91 – 104.
- [12] COBURN J, CAULFIELD A M, AKEL A, et al. NV-Heaps: making persistent objects fast and safe with next-generation, non-volatile memories[J]. ACM SIGPLAN Notices, 2011, 46(3): 105 – 118.
- [13] HAHN S. A protected block device for persistent memory[C]// Proceedings of the 2014 30th Symposium on MASS Storage Systems and Technologies. Piscataway, NJ: IEEE, 2014: 1 – 12.
- [14] LU Y, SHU J, SUN L, et al. Loose-ordering consistency for persistent memory[C]// Proceedings of the 2014 IEEE 32nd International Conference on Computer Design. Piscataway, NJ: IEEE, 2014: 216 – 223.
- [15] BHANDARI K, CHAKRABARTI D R, BOEHM H J. Implications of CPU caching on byte-addressable non-volatile memory programming[EB/OL]. [2016-06-20]. <http://hpl.hp.com/techreports/2012/HPL-2012-236.pdf>.
- [16] Intel64 and IA-32 architectures software developer's manuals combined[EB/OL]. [2016-10-08]. <http://www.intel.com>.
- [17] PHILLIPS D. A directory index for ext2[C]// Proceedings of the 5th Annual Linux Showcase & Conference. Berkeley, CA: USENIX Association, 2001: 20.
- [18] REISER H. ReiserFS v.3 Whitepaper[EB/OL]. [2016-10-08]. <http://web.archive.org/web/20031015041320>.
- [19] SEHGAL P, BASU S, SRINIVASAN K, et al. An empirical study of file systems on NVM[C]// Proceedings of the 2015 31st Symposium on Mass Storage Systems and Technologies. Piscataway, NJ: IEEE, 2015: 1 – 14.
- [20] SHA H M, CHEN X, ZHUGE Q, et al. Designing an efficient persistent in-memory file system[C]// Proceedings of the 2015 IEEE Non-Volatile Memory System and Applications Symposium. Piscataway, NJ: IEEE, 2015: 1 – 6.
- [21] BSDDirhash[EB/OL]. [2016-10-08]. <http://groups.yahoo.com/group/freebsd-hackers/message/62664>.
- [22] dir-index-pmfs[Z/OL]. [2016-10-08]. <https://github.com/dir-index-pmfs/dir-index-pmfs>.

This work is partially supported by the National High Technology Research and Development Program (863 Program) of China (2015AA015305).

YANG Shun, born in 1992, M. S. candidate. His research interests include non-volatile memory, file system.

CHEN Zhiguang, born in 1984, Ph. D., assistant researcher. His research interests include high performance storage, high performance computing.

XIAO Nong, born in 1969, Ph. D., professor. His research interests include high performance computing, large-scale network storage.

(上接第1240页)

- [20] NIEME D C. Unixbench 4.1.0 [CP/OL]. [2016-06-20]. <http://www.tux.org/pub/tux/niemi/unixbench>.

This work is partially supported by the National High Technology Research and Development Program (863 Program) of China (2015AA015307), the National Natural Science Foundation of China (61472323), the Fundamental Research Funds for the Central Universities (3102015JSJ0009).

LIU Chunyi, born in 1993, M. S. candidate. His research interests include cloud computing, cloud storage.

ZHANG Xiao, born in 1978, Ph. D., associate professor. His research interests include cloud computing, cloud storage, mass storage.

LI Ani, born in 1992, M. S. candidate. Her research interests include cloud computing, cloud storage.

CHEN Zhen, born in 1992, M. S. candidate. His research interests include cloud storage.