

云存储性能评测研究

周小鹏 张小芳 赵晓南

(西北工业大学计算机学院 西安 710129)

摘要 大数据时代的到来,使得对云存储产生了强烈的需求。云存储性能评测一直是云存储系统研究的热点与难点。通过分析典型云存储平台数据访问的过程,建立云存储数据读写的模型,利用网络分析工具 iptools 获得云存储系统在处理用户访问请求的各个阶段的时间开销,从中排除网络干扰,反映真正云存储系统数据读写性能。实验结果表明,这种云存储系统数据读写模型是科学合理的,按照模型设计的云存储系统性能评测方法能够有效排除网络干扰,获取反映云存储系统真正读写性能的数据。

关键词 云存储,性能评测,评测指标

中图分类号 TP399 文献标识码 A

Research of Performance Evaluation of Cloud Storage

ZHOU Xiao-peng ZHANG Xiao-fang ZHAO Xiao-nan

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

Abstract After the coming of Big Data Time, the demands for cloud storage are becoming stronger. Performance evaluation of cloud storage is always the key and difficult point of research of cloud storage system. Through analyzing the data access process of typical cloud storage platforms, the data access model of cloud storage was built. With the help of iptools capturing IP packages, the time cost of each phase of processing users' access was gained. The true performance can be observed when ignoring the network interference. The test results show that the model is scientific and the test method can gain the true performance of cloud storage system effectively.

Keywords Cloud storage, Performance evaluation, Evaluation metrics

1 引言

大数据时代的到来,使得对数据基础设施的需求极为旺盛。根据 IDC 的统计数据,2005 年全球数据总量为 1300 亿 GB,2010 年达到 12270 亿 GB,到了 2012 年又翻了两倍,预计 2020 年是 2010 年的 30 倍之多(约 40ZB)。一味地增加存储容量并不符合各个组织的利益诉求,云存储的出现解决了这一难题。云存储一方面满足了组织的数据存储需求,另一方面为云存储服务提供商带来了利益,这种双赢模式也是云存储一经提出就异常火热的原因。现在的互联网用户不再满足于本地存储,需要随时、随地访问数据资源。传统的存储方式已不能满足用户的这种需求,云存储却能更好地为用户提供这类服务。宽带接入、Web2.0、分布式存储和存储虚拟化等技术的出现,使得云存储从需求、理论变成了现实。目前云存储产品多种多样,各有特点,例如,Amazon S3、Google Drive、Microsoft SkyDrive、Dropbox、Apple iCloud 等等,国内许多互联网企业也相继推出自己的云存储产品和开放平台,例如 Baidu PCS、金山 Kuaipan、360 网盘和 115 网盘等等。如此多的云存储产品,给用户的选择造成了很大的困惑。如何选择

云存储产品,数据读写性能、数据安全性和使用方便等都是要考虑的因素,但是最主要的还是考察其数据读写性能,这是数据存储的关键所在,也是云存储领域研究的热点和难点。在国家 863 项目“云计算关键技术及系统”中,本单位承担的云存储系统评测从多个方面对云存储进行全方位的测试,其中性能测试将是研究重点和难点,本文将提出的测试指标和模型将会是该项目云存储性能评测部分的关键。

2 研究内容及现状

2.1 传统存储系统性能评测研究

存储系统性能评测的关注点历来都是集中在评测指标和评测方法上,根据预先确立的评测指标,针对不同的评测对象,设计相应的评测方法进行性能测试。传统存储系统的性能评测指标主要有 5 个:聚合带宽(Bandwidth)、吞吐量(Throughput)、并发连接数、响应时间(Response Time)和 IOPS。聚合带宽指的是系统允许的最大数据传输速度,使用压力测试,探寻系统的极限。并发连接数衡量存储系统同一时刻允许多少个正常 IO 访问。吞吐量指的是一定时间内被系统处理的数据量。IOPS 指的是系统单位时间内处理的 IO

到稿日期:2013-06-06 返修日期:2013-08-30 本文受西工大基础研究基金(JC20120209),国家科技支撑计划课题(2011BAH04B05),国家 863 重大项目(2013AA01A215),自然科学基金重点项目(61033007)资助。

周小鹏 男,硕士生,主要研究方向为网络存储、海量存储、云计算、云存储;张小芳 女,博士,副教授,主要研究方向为软件工程、数据容灾、工作流建模技术、分布式计算技术;赵晓南 女,博士,讲师,主要研究方向为海量存储系统管理、评测的理论与技术、海量数据管理技术等。

次数,对随机访问型应用,该指标比较重要。响应时间(Response Time)是指从应用发出 IO 请求到存储系统返回有效数据为止的整个时间,可以从整体上衡量存储系统的性能,是评测存储系统性能的关键性指标。

2.2 云存储系统性能评测研究

针对云存储性能的研究可以分为横向和纵向两个方面。横向确定不同的评测指标,纵向确定云存储性能研究的层次。云存储是传统存储的延续与发展,衡量传统存储系统性能的一些指标也可以适用于云存储系统。虽然吞吐量可以衡量云存储系统的整体 IO 性能,但是由于云存储的特殊性(通过网络连接),吞吐量很大程度上受网络传输能力的影响。由于在云存储数据传输过程中,将客户端每次 IO 请求的所有数据都放置到与云存储服务端连接的流(Stream)中,因此针对云存储系统测试 IOPS 就没有了实际意义。响应时间可以从发出数据访问请求到此次访问完成整个过程来衡量云存储系统完成一次 IO 的效率,并且能够划分成不同的时间段,以便深入地分析影响云存储系统性能的各个因素。不同的人员对云存储系统的研究层次是不同的。云存储系统管理人员可能更关心系统整体性能,系统设计人员考虑最多的可能是虚拟设备、接口服务节点和服务管理节点的性能,所以明确评测层次非常重要。每个层次的性能评测对云存储的研究意义重大,本文所提的测试模型是站在普通用户的角度,在不明确云存储系统内部设计以及其运作过程的前提下,从云存储系统外部客观评测系统性能。

2.3 云存储性能研究现状

目前针对云存储系统性能的评测,从理论到实践,并不是很多。文献[1]提出了一套开源的 benchmark,分析了 Azure 云平台存储服务的综合性能。文献[2]基于 CAP 定理和 No-SQLLogic 云系统 CAP 性能评估框架,提出一种性能评测方法。针对 Amazon SimpleDB 和 Windows Azure Table Storage,提供了庞大的测试集。文献[3]通过整合 Azure 分布式 Cache 技术和 Table Storage 的数据完整性技术,提出一种中间存储服务——HCSF(混合云存储架构)。文献[4]详细分析和论述了 Amazon SimpleDB 和 Google Big Table Database 的特征、框架和优点,从而给用户选择云端数据库提供了参考。文献[5]通过对 Dropbox 进行带宽密集型 and 事务密集型混合任务的测试,发现了系统在云中的虚拟机间同步的极大延迟问题。

无疑,在这些研究中并没有考虑连接云存储用户端和云存储服务器的底层网络对云存储性能的影响。用户本地网络严重影响到用户体验,用户端表现出的性能并不能真正地反映云存储服务提供商的服务水平。本文基于此提出自己的云存储性能评测模型,并以系统响应时间(Response Time)为评测指标,对云存储系统性能进行客观的评测,以排除网络对云存储性能评测的影响。

3 评测模型

虽然各个云存储厂商为用户提供了丰富多样的功能,但是对普通用户来讲,这些功能归结起来就是读(下载)和写(上传),即云存储为用户提供的两项基本功能。本文通过测试云

存储系统在用户写(上传)、读(下载)数据的响应时间来达到观测系统性能的目的。

3.1 写(上传)性能评测

写过程分析 通过分析 Baidu PCS、金山快盘、Dropbox 等云存储产品的数据上传过程(如图 1 所示),将用户写(上传)数据至云存储服务器的过程分为 5 个部分:写请求阶段、写准备阶段、数据传输阶段、系统存储阶段、写响应阶段。

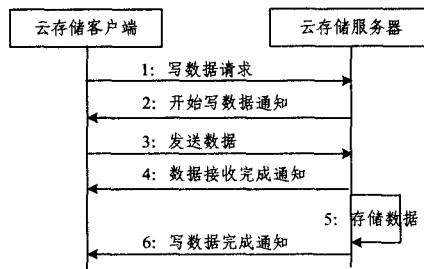


图 1 写(上传)数据过程

写过程建模 根据上述用户上传(写)数据过程的划分,将其相应的总时间开销也进行进一步的分割。

$$TC_{up} = TC_{req} + TC_{pre} + TC_{trans} + TC_{store} + TC_{resp} \quad (1)$$

式中, TC_{up} 表示上传(写)数据总的的时间, TC_{req} 表示用户发出上传数据请求至服务端允许开始上传数据这段时间, TC_{pre} 表示客户端上传数据的准备时间(这段时间很少,几乎可以忽略不计), TC_{trans} 表示数据在网络中的传输时间, TC_{store} 表示云存储系统将数据写到其数据中心的时间开销(从接受完所有数据到写完成这段时间), TC_{resp} 表示云存储系统把此次存储结果反馈给客户的时间开销。其中 TC_{trans} 在整个开销中占的比重最大, TC_{req} 可以衡量云存储系统响应用户上传(写)数据请求的能力,而 TC_{store} 对评价一个云存储系统的性能最为关键,反映系统内部处理用户数据存储的能力。

本文假定客户端发送数据比网络传输快,以目前的主机处理水平, TC_{pre} 是非常小的,可以忽略不计。 TC_{resp} 基本都花费在客户端与服务端的确认通信上,对系统性能评测的影响微乎其微,在测试中忽略不计。由于云存储系统响应用户写请求的能力不是本文研究的重点,因此在以后的实验部分不再涉及 TC_{req} 。本文只关注式(1)中 TC_{trans} 和 TC_{store} 两部分。本文利用 iptools 工具捕获云存储客户端与云存储服务器端所用通信 ip 报文(包括数据报文),通过分析上传数据过程中的 ip 报文信息,将 ip 报文按照上传数据过程的阶段划分归类,计算出各个阶段的时间开销。

3.2 读(下载)性能评测

读过程分析 用户从云端读(下载)数据的过程(见图 2)可以划分为 4 个阶段:读请求阶段、服务端数据准备阶段、数据传输阶段、读响应阶段。

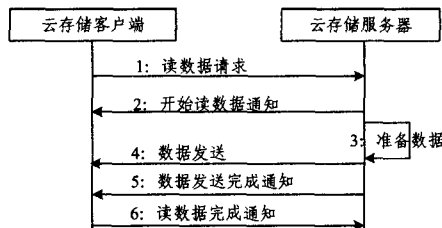


图 2 读(下载)数据过程

读过程建模 根据图 2 和读(下载)数据的过程划分,本文将云存储系统中读数据的总耗时按式(2)分段。

$$TC_{down} = TC_{req} + TC_{pre} + TC_{trans} + TC_{resp} \quad (2)$$

式中, TC_{down} 是下载数据总的时间开销, TC_{req} 表示客户发起读数据请求至云存储服务器开始发送数据这段时间, TC_{pre} 表示服务端发送数据前的准备时间, TC_{trans} 是数据在网络中的传输开销, TC_{resp} 表示数据下载完成后云存储服务端响应的的时间开销。其中, TC_{req} 可以衡量一个云存储系统响应用户下载数据请求的能力, TC_{pre} 表示从云存储系统真正读取数据的能力, 而 TC_{trans} 对用户的访问性能影响最为显著。其中云存储系统处理用户读请求的能力不是本文所关心的, 所以在以后的测试中不涉及 TC_{req} 。 TC_{resp} 读请求响应不影响对云存储系统读性能的评测, 时间比较短, 主要花费在读完数据后客户端和服务端的控制通信, 在后续测试中不予考虑。

针对云存储系统下载数据性能的评测, 本文假定客户端从网络接收数据的能力大于网络传输数据的能力, 云存储客户端接收数据能力不能成为从云端下载数据的瓶颈。在下载数据的过程中, 利用 iptools 工具捕获云存储客户端与服务端的所有 ip 报文, 按照下载请求、数据准备、数据传输和服务响应划分归类, 计算出云存储服务端数据准备的时间开销 TC_{pre} 和数据在网络中的传输开销 TC_{trans} 。云存储服务端数据准备的时间开销可以反映该云存储系统内部读取数据的性能, 是评价云存储系统的关键指标。数据在网络中传输的时间开销 TC_{trans} 反映数据下载过程中的网络状况。

4 实验

4.1 实验设计

针对云存储上传、下载性能的测试, 本文选择 Baidu PCS^[6] 和金山 Kuaipan^[7] 作为测试对象, 上述两个云存储产品都提供了用于编程的 API, 从其官网可以获得相应的 SDK。为了使测试更加公平公正, 将测试端程序分别部署到同一局域网中的两台主机上, 同时使用 iptools 网络抓包工具捕获测试端与云存储服务端的所有 ip 报文。

写(上传)实验设计。本文针对不同的数据大小, 测试了 Baidu PCS 和金山 Kuaipan 写性能。选取的数据大小分别是 512kB、1MB、2MB、4MB、8MB, 在同一数据大小的测试中选取了若干个采样点, 分别测试此时的 TC_{trans} 、 TC_{store} 。将测得的数据在同一云存储产品内部和不同云存储产品间进行比较。获得云存储产品性能随数据大小的变化情况, 以及不同云存储产品数据上传性能的差异。测试实验安排如表 1 所列。

表 1 写(上传)测试次数

	PCS	Kuaipan
512k	50	50
1M	50	50
2M	50	50
4M	50	50
8M	30	30

在每一次测试中用到的测试文件数据是随机产生的, 将这些文件通过 Baidu PCS 和 Kuaipan 提供的写数据接口上传到它们的服务器中, 获取这一过程中的 TC_{trans} 和 TC_{store} 。

读(下载)实验设计。在测试 Baidu PCS 和金山 Kuaipan 读性能的时候, 观测不同大小的数据上传过程中网络的传输时间 TC_{trans} 和系统的读准备时间 TC_{pre} 。产生 512k、1M、2M、4M、8M 大小的随机文件, 将这些文件首先上传至 Baidu PCS 和金山 Kuaipan 中, 为读(下载)做准备, 然后以数据大小进行分组测试, 测试安排如表 2 所列。

表 2 读(下载)测试次数

	PCS	Kuaipan
512k	50	50
1M	40	40
2M	40	40
4M	40	40
8M	40	40

4.2 实验环境

根据 4.1 节的实验设计, 本文进行的云存储系统性能评测实验测试端部署在同一局域网内的两台主机之上, 它们的各项配置如表 3 所列。

表 3 测试端配置信息

	硬件	软件
Baidu PCS	CPU Intel(R) Core i5, 内存 4GB, 千兆以太网网卡	Win7, eclipse3.7, java1.7, bcs-sdk-java_1.4.3.jar, iptools
Kuaipan	CPU Intel(R) Core i5, 内存 4GB, 千兆以太网网卡	Win7, eclipse3.7, java1.7, com.kuaipan.jdk.jar, iptools

本次测试的云存储对象分别是 Baidu PCS 和金山 Kuaipan, 在测试开始前向 Baidu 和金山开发平台申请了开发者账户, 可以通过它们提供的 API 访问其云存储产品, 有关它们提供的云存储服务信息如表 4 所列。

表 4 待测云存储产品信息

	Baidu PCS	Kuaipan
总容量	20G	7G
允许流量(月)	60G	没有限制
允许请求数(月)	200W	没有限制
允许的 Bucket	5	没有限制

4.3 结果分析

4.3.1 写(上传)性能测试结果分析

以写(上传)数据大小 512k、1M、2M、4M、8M 为分组标准, 测得各组的 TC_{trans} 和 TC_{store} , 计算各个分组内 TC_{trans} 和 TC_{store} 的期望和方差。 TC_{trans} 期望反映在数据大小确定的情况下网络传输所表现出的一边状况, TC_{store} 期望反映在数据大小确定时云存储系统内部写(存储)数据到自己的数据中心的平均效率。 TC_{trans} 方差是指网络传输时间围绕其均值的波动情况, 反映的是在数据传输过程中网络传输的稳定性; TC_{store} 方差是指云存储系统内部存储数据的时间在其均值周围波动的情况, 反映了云存储系统向其数据中心写数据的稳定性。

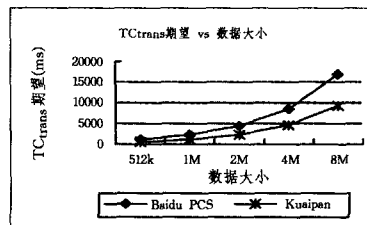


图 3 写数据 TC_{trans} 期望 vs 数据大小

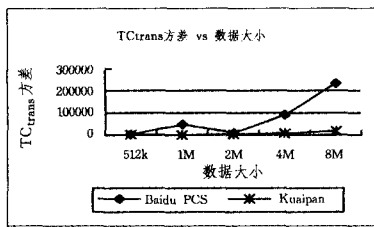


图4 写数据 TC_{trans} 方差 vs 数据大小

从图3可以看出,随着上传数据的增大,不管是PCS还是Kuaipan,它们的网络传输时间开销都在增加,但是并没有呈直线型增加,这是由于在网络传输数据报文的过程中,随着数据量的增大,用于控制传输的报文增多,在长连接的情况下,丢包数据也会增大,因此网络传输时间并没有随数据大小呈现线性增长,而表现出指数型增长。结合图4,数据在网络传输过程中,金山Kuaipan TC_{trans} 的方差都低于Baidu PCS的 TC_{trans} 的方差,这说明在传输数据的过程中,连接测试端和Kuaipan的网络稳定性要优于Baidu PCS。

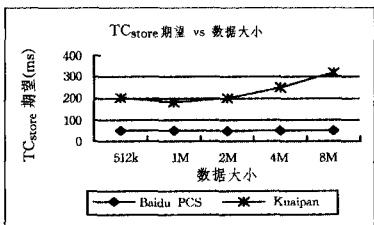


图5 写数据 TC_{store} 期望 vs 数据大小

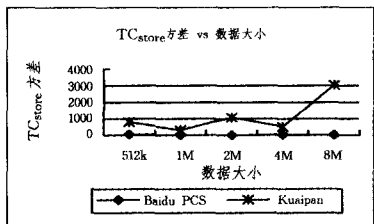


图6 写数据 TC_{store} 方差 vs 数据大小

从图5和图6可以看出Baidu PCS向自己的数据中心写数据的时间开销主要集中在50ms周围,并没有随着数据大小改变而发生明显的变化。Kuaipan向数据中心写数据的时间开销随着数据的增大而增大,系统存储数据的时间开销与数据大小呈现较强的正相关,只是在512k至2M这个阶段出现了一定程度的下降,经分析发现其原因是Kuaipan系统内部在为小数据分配存储空间时花费的时间有所差异,在数据量增加到一定程度之后,Kuaipan系统内写时间开销随数据量的增加表现出明显的上升趋势。如图6所示,Baidu PCS系统存储时间开销变化不及Kuaipan的变化,表现得很稳定。

通过分析ip报文,发现Baidu PCS网络传输时间开销远远大于Kuaipan的原因是,Kuaipan上传数据至本市(西安市)的存储中心,而Baidu PCS需要上传数据至北京的存储中心,这可能是引起Baidu PCS网络传输开销的主要原因。Baidu PCS在存储数据时,可能采用一边从网络接收数据一边存储的方式,云存储服务接收数据的能力通常优于网络传输,这样使得存储时间开销没有随着数据大小的变化而发生明显的变化;而Kuaipan的系统存储时间随数据大小有明显的变化。从总体上来看,Baidu PCS的系统存储时间远远小

于Kuaipan的系统存储时间。由此可以得出,在排除网络传输的影响之后,Baidu PCS存储用户数据的性能优于Kuaipan。

4.3.2 读(下载)性能测试结果分析

从云端下载数据到客户端的过程就是用户从云中读数据的过程,主要影响用户读性能的因素集中在网络传输部分和云存储服务从其数据中心读取数据这部分。本次针对Baidu PCS和金山Kuaipan读性能的测试中,从云端读取512k、1M、2M、4M、8M不同大小的数据,不同大小的数据读取若干次(具体说明如4.1节),分析读取同一大小数据过程中 TC_{pre} (云存储服务内部读数据时间)和 TC_{trans} (网络传输时间)取得的各个小组的 TC_{pre} 期望、方差和 TC_{trans} 期望、方差,从平均水平反映云存储内部读数据性能及连接云存储和客户端的网络性能。

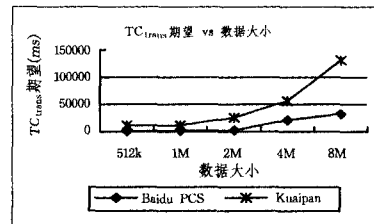


图7 读数据 TC_{trans} 期望 vs 数据大小

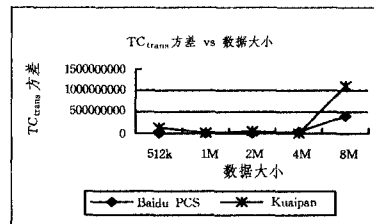


图8 读数据 TC_{trans} 方差 vs 数据大小

从图7可以看出,Baidu PCS和Kuaipan在下载数据过程中,网络传输时间随数据大小的增加都呈现出增加的趋势。结合图8,在读(下载)数据的过程中,Kuaipan网络传输时间波动性比Baidu PCS大。在整体上,Baidu BCS的网络传输时间远小于Kuaipan的网络传输开销,这与Baidu在读(下载)数据的过程中维护多条TCP长连接有一定的关系。

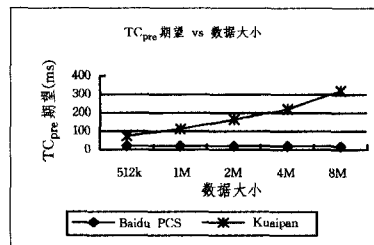


图9 读数据 TC_{pre} 期望 vs 数据大小

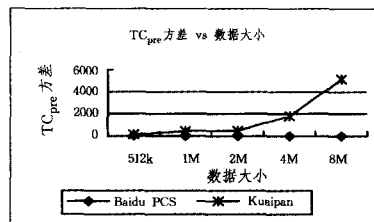


图10 读数据 TC_{pre} 方差 vs 数据大小

在图9中,Baidu PCS的数据准备时间 TC_{pre} 随数据大小的变化不甚明显,在18~22ms周围波动,而Kuaipan的 TC_{pre} 随着数据大小的增加有明显的增加趋势,并且从图10可以看出,Kuaipan的数据准备时间 TC_{pre} 波动性远远大于Baidu PCS,Baidu PCS的 TC_{pre} 围绕其均值做小范围的变动。Baidu PCS TC_{pre} 随数据大小变化不明显的原因是Baidu PCS在读取若干数据块之后就发送大网络,而Kuaipan则是在读完整个数据之后才发送到网络上,所以Baidu PCS表现出 TC_{pre} 变化不明显,而Kuaipan的 TC_{pre} 表现出随数据大小增大而增大的情况。

综上,在读数据的过程中,Baidu PCS不论是在网络传输 TC_{trans} 还是在数据准备 TC_{pre} 上均优于Kuaipan,并且其稳定性也优于Kuaipan。在排除网络引起的干扰之后,Baidu PCS在读取用户数据时的性能要优于Kuaipan。

结束语 本文通过分析云存储客户端与云存储服务通信IP报文,找出读写数据的几个关键时间点,计算出读写数据各个阶段所用时间,分析这些时间开销与数据大小的关系,比较了Baidu PCS和金山Kuaipan之间对应时间开销的大小,得出在不考虑网络带宽的情况下,Baidu PCS的性能优于金山Kuaipan。从实验结果可以看出,本文提出的读写时间开销模型能够很好地评测不同的云存储系统。可以根据侧重点不同,使用本文所提模型观测影响云存储系统的各个因素,对云存储进行完整的评测。

(上接第167页)

参 考 文 献

- [1] Buneman P, Khanna S, Wang C T. Why and where: A characterization of data provenance[C]//Proceedings of the International Conference on Database Theory (ICDT). 2001
- [2] Callahan S, Freire J, Santos E, et al. VisTrails: Visualization meets data management[C]//Proceedings of ACM SIGMOD International Conference on Management of Data (SIGMOD). 2006
- [3] Liu M, Taylor N E, Zhou W, et al. Recursive Computation of Regions and Connectivity in Networks[C]//ICDE. 2009
- [4] Zhou W, Sherr M, Tao T, et al. Efficient querying and maintenance of network provenance at internet-scale[C]//SIGMOD. 2010
- [5] Karvounarakis G, GIVES Z, Tannen V. Querying data provenance [C]//SIGMOD'10 Proceedings of the 2010 International Conference on Management of Data. 2010:951-962
- [6] Green T J, Karvounarakis G, Taylor N E, et al. ORCHESTRA: Facilitating collaborative data sharing[C]//Proceedings of ACM SIGMOD International Conference on Management of Data (SIGMOD). 2007
- [7] Cohen-Boulakia S, Biton O, Cohen S, et al. Addressing the prove-

参 考 文 献

- [1] Agarwal D, Prasad S K. Azurebench. Benchmarking the storage services of the Azure Cloud Platform[C]//Parallel and Distributed Processing Symposium Workshops & PhD Forum (IP-DPSW), 2012 IEEE 26th International. IEEE, 2012:1048-1057
- [2] Benefico S, Gjerci E, Gomasasca R G, et al. Evaluation of the CAP Properties on Amazon SimpleDB and Windows Azure Table Storage[C]//Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), 2012 14th International Symposium on. IEEE, 2012:430-435
- [3] Neumann R, Taggeselle S, Dumke R, et al. Combining Query Performance with Data Integrity in the Cloud: A Hybrid Cloud Storage Framework to Enhance Data Access on the Windows Azure Platform[C]//Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on. IEEE, 2012:518-525
- [4] Ramanathan S, Goel S, Alagumalai S. Comparison of Cloud database: Amazon's SimpleDB and Google's Bigtable[C]//Recent Trends in Information Systems (ReTIS), 2011 International Conference on. IEEE, 2011:165-168
- [5] Wang H, Shea R, Wang F, et al. On the impact of virtualization on Dropbox-like cloud file storage/synchronization services[C]//Proceedings of the 2012 IEEE 20th International Workshop on Quality of Service. IEEE Press, 2012:11
- [6] Baidu PCS. <http://developer.baidu.com/wiki/index.php?title=docs/pcs>
- [7] 金山 Kuaipan. <http://www.kuaipan.cn/developers/document.htm>

nance challenge using zoom[J]. Concurrency and Computation: Practice and Experience, 2008, 20:497-450

- [8] Ikeda R, Park H, Widom J. Provenance for generalized map and reduce workflows[C]//Proceedings of Biennial Conference on Innovative Data System Research (CIDR). 2011
- [9] System W, Altintas I, Barney O, et al. Provenance collection support in the kepler scientific workflow system[C]//Proceedings of the International Provenance and Annotation Workshop (IPAW). 2006
- [10] Hasan R, Sion R, Winslett M. Preventing history forgery with secure provenance[J]. ACM Transactions on Storage (TOS), 2009, 5(4):1-43
- [11] Green T J, Karvounarakis G, Tannen V. Provenance semirings [C]//Proceedings of the ACM Symposium on Principles of Database Systems(PODS). 2007
- [12] Network Simulator 3[OL]. <http://www.nsnam.org/>
- [13] Liu X, Guo Z, Wang X, et al. D3S: Debugging Deployed Distributed Systems[C]//NSDI. 2008
- [14] RapidNet[OL]. <http://netdb.cis.upenn.edu/rapidnet/>
- [15] Loo B T, Condie T, Garofalakis M, et al. Declarative Networking [C]//CACM. 2009
- [16] 万亚平, 冯丹, 欧阳利军, 等. 一种适用于 P2P 存储系统的自反馈故障检测算法[J]. 计算机科学, 2010, 37(2):48-52