

# 近年来 Hadoop 国外研究综述<sup>①</sup>

王彦明<sup>1</sup>, 奉国和<sup>2</sup>, 薛云<sup>3</sup>

<sup>1</sup>(河南中医学院 郑州 450008)

<sup>2</sup>(华南师范大学 经济与管理学院, 广州 510006)

<sup>3</sup>(华南师范大学 物理与电信工程学院, 广州 510006)

**摘要:** 综合论述近年来 Hadoop 国外研究成果, 从论文发表年代, 来源国家, 作者单位类型, 论文来源企业和主题等方面进行文献计量分析, 从理论研究和应用研究两方面进行内容分析, 理论研究包括比较研究, 任务调度, 性能优化和功能扩展等, 应用研究包括云存储, 数据查询, 数据分析, 数据挖掘, 关联广告等, 希望以此为进一步研究提供参考。

**关键词:** 云计算; Hadoop; MapReduce

## Summary of Hadoop Research in Recent Years in Foreign Countries

WANG Yan-Ming<sup>1</sup>, FENG Guo-He<sup>2</sup>, XUE Yun<sup>3</sup>

<sup>1</sup>(Henan University of TCM, Zhengzhou 450008, China)

<sup>2</sup>(School of Economics and Management, South China Normal University, Guangzhou 510006, China)

<sup>3</sup>(School of Physics and Telecommunication Engineering, South China Normal University, Guangzhou 510006, China)

**Abstract:** This paper discusses comprehensively foreign Hadoop researches in recent years. Literature metrological analyses include publishing years, source countries, the author's organization type, topic, etc. Content analysis includes theory and applied research. Theoretical researches include comparative, task scheduling and efficiency improvement. Application researches include cloud storage, searching, data analysis, data mining, and relevance advertising. It is hope that some reference can be used for forward research.

**Key words:** cloud computing; Hadoop; Mapreduce

近年来, 随着信息技术的飞速发展, 各种网络应用带来了数据规模的高速增长, 为了满足海量数据存储和分析需求, 需要使大量计算机协同工作共同完成空前复杂的任务。Hadoop 是 Apache 软件基金会旗下的一个开源分布式计算平台, 它基于 Java 语言开发, 核心是 HDFS 和 MapReduce。HDFS 具有高容错性和高扩展性等优点, 允许用户将 Hadoop 部署在价格低廉的服务器上, 形成分布式系统; MapReduce 分布式编程模型允许用户在不了解分布式系统底层细节的情况下开发并行应用程序。因此通过 Hadoop 可以轻松地组织计算机资源, 搭建自己的分布式计算平台, 完成海量数据的处理。相对当前应用较多的

SQL 关系型数据库, HDFS 提供了一种通用的数据处理技术, 它用大量低端服务器代替大型单机服务器, 用键值对代替关系表, 用函数式编程代替声明式查询, 用离线批量处理代替在线处理; 作为编程模型的 MapReduce 将分布式编程分为 Map(映射) 和 Reduce(化简)两个阶段, 基于 MapReduce 模型编写的分布式程序可以将一项任务分发到上千台商用机器组成的集群上, 以高容错的方式并行处理大量的数据集。HDFS 和 MapReduce 共同组成了 Hadoop 分布式系统体系结构的核心, 共同完成分布式集群的计算任务。目前互联网领域的 Web 搜索、广告系统、数据分析和机器学习等许多任务已经在 Hadoop 集群上

① 基金项目:广东省科技计划工业攻关项目(2009B010800051);广东省现代信息服务业发展专项资金扶持项目(GDEID2010IS034);广州市科技计划项目(2011J4300046)

收稿时间:2012-04-13;收到修改稿时间:2013-06-18

运行. 本文通过广泛分析 Hadoop 在众多领域的应用, 进行一定程度的分类和归纳, 希望据此为各行业的程序员提供一些参考.

### 1 文献计量分析

笔者在 ACM 数字图书馆中以 Hadoop 为主题检索获得 218 篇论文, 发表年份分布在 2007 年到 2011 年之间, 统计结果见表 1, 从发展趋势上看 2008 年以后随着云计算的兴起, Hadoop 作为最重要的开源云计算技术成为各方面的研究热点.

表 1 论文发表年份

	2007	2008	2009	2010	2011	合计
数量 (篇)	3	18	43	114	40	218
比例 (%)	1.38	8.26	19.72	52.29	18.35	100

从地域分布来看美国约占 76.15%, 这是因为推动云计算产生和发展的企业象 Google、雅虎、微软等主要来自美国. 其他国家研究较少合计仅占 16.06%. 详细分析见表 2.

表 2 论文来源国家

国家	美国	德国	中国	韩国	希腊
数量 (篇)	166	9	6	4	2
国家 (篇)	奥地利	日本	瑞士	加拿大	新加坡
数量 (篇)	1	1	2	3	2
国家 (篇)	意大利	英国	法国	以色列	国际协作
数量 (篇)	1	1	2	1	17

从论文作者单位来看研究者主要在高校和企业, 来自高校的论文占 55.5%, 来自企业的占 33.03%, 两方面合作撰写的占 11.47%. 这表明 Hadoop 在产业界和学术界都得到普遍重视. 具体统计见表 3.

表 3 作者所在机构类型

来源	高校	企业	校企合作	合计
数量 (篇)	121	72	25	218
比例 (%)	55.5	33.03	11.47	100

文献统计表明, 对 Hadoop 进行研究的高校中美国的加利福尼亚大学、马里兰大学、伊利诺斯州立大学和休斯顿大学等发表论文较多, 德国的 RWTH Aachen 大学, 韩国汉城国家大学, 新加坡国立学院, 希腊雅典大学, 英国格拉斯哥大学等也有一些研究成果. 企业中主要有雅虎、微软、Google、IBM、Facebook 和惠普等, 其他机构还有一些数据分析公司、研究所和技术协会等. 具体统计见表 4.

表 4 论文来源企业统计

	雅 虎	微 软	Google	IBM	Facebook	HP	易 趣	AT&T	新 浪	合 计
自主研 究	19	9	4	3	2	2	0	0	0	39
合作研 究	12	5	5	4	3	2	2	1	1	35

从研究内容上分析总体上可以分为理论和应用两大方面. 从统计结果上看理论研究占 42.2%, 应用研究占 57.8%, 理论研究主要是性能优化和任务调度, 应用研究主要是数据分析和数据查询. 具体见表 5.

表 5 论文主题统计

理论研究	比较 研究	任务 调度	性能 优化	功能 扩展	合计	
数量 (篇)	9	26	36	21	92	
比例 (%)	9.78	28.26	39.13	22.83	100	
应用研究	云存储	数据 查询	数据 分析	数据 挖掘	关联 广告	合计
数量 (篇)	12	38	47	13	16	126
比例 (%)	9.52	30.16	37.3	10.32	12.7	100

### 2 Hadoop理论研究

#### 2.1 比较研究

Hadoop 是 Google 云计算技术的开源实现, 主要用于处理大规模非结构化数据. 传统数据处理多采用关系数据库系统, 关系数据库擅长结构化数据处理, 经过多年的发展已经具有广阔的应用范围. 但是由于业务发展和数据规模的快速增加, SQL 等传统的关系数据库在查询效率上逐渐不能满足需求, 而且建设和维护的成本高. MapReduce 擅长对半结构化或非结构化数据进行复杂的分析, 容错能力强而且其基础设施可以灵活扩展. 一些企业采用并行关系数据库提高数据处理速度, 但是并行 RDBMS 系统设计和建设困难, 不及 Hadoop 灵活, 建设成本高, 而且容错能力不足. 大量非结构化数据分析应用表明 Hadoop 与目前两种主要的并行 DBMS 相比速度大约快 3-6 倍. 例如 Clustera 是基于关系数据库的集群管理系统, 具有良好的扩展性能, 但是在大规模非结构化数据处理方面性能远不及 Hadoop 效率高<sup>[1]</sup>. 除了应用类型方面的比较外研究者还将 Hadoop 与同类开源云计算系统进行了比较. 例如 Sector/Sphere 是一个类似 Hadoop 的开源云计算系统, 它以分布式文件系统 Sector 为基础进行

数据密集型计算,使用一个叫做 UDT 的高速传输协议,支持用户定制的管理功能.它不按照块大小确定任务数,多在广域网上跨越多个数据中心部署.

## 2.2 任务调度研究

在 Hadoop 中一项工作被分割成许多小块,这些小块叫做任务.不同的任务被分配到不同节点上进行计算,因此分布式系统设计成功的关键是建立一种高效的调度机制.这方面需要解决两个问题,一是如何将任务优先分配给空闲的机器,使所有的任务公平分享系统资源;二是如何尽量将 map 任务分配给 InputSplit 所在的机器,目的是减少不同节点之间的数据传输.一种有效的方法是指定动态优先权,随时间管理节点并指定优先权可以自动地侦测和消除一项工作的瓶颈,给任务提供均衡的计算能力以适应不同阶段的需求<sup>[2]</sup>.细粒度资源共享是分布式系统任务调度的重要策略,它将任务调度问题转换为节点位置公平性求解,通过对数据位置权重和节点负载进行衡量,根据总体优化模型进行均衡配置以达到最佳任务分配.分布式控制器<sup>[3]</sup>可以在工作执行过程中持续监控和调整计算资源,调节不同节点任务的运行进度,提供一项工作处理的并发子任务加速或减速能力,使得这项任务在与其他共享资源节点的任务的完成时间可以被预期,从而进行合理的弹性资源分配.有些研究采用抽象提炼方法进行任务调度<sup>[4]</sup>,这种方法首先提取工作和计算资源作为抽象代表进行安排,作为抽象的代表通常比较小,时序安排可以很快完成,如果不能满足需求,就会定义工作和数据进一步提炼直到满足各阶段任务调度要求为止.

## 2.3 性能优化研究

Hadoop 采用的是 Google 最初提出的 MapReduce 模型,在具体应用场景中会面临许多效率方面的问题,为此程序设计者不断对算法进行改进以提高其执行效率.这些改进表现在解决数据不均衡问题、管理中间数据、治理 I/O 冗余等.Hadoop 对任务的不同分割依赖于不同的输入规模,并且非周期性动态变化,表现出严重的数据不均衡.SkewReduce 系统<sup>[5]</sup>用于解决科学计算中进行用户定制特征提取工作的计算数据不均衡问题,系统的核心是一个优化器,通过在表达特征时设置参数可以最好地分割输入的数据,将计算不均衡降到最低.Scarlett<sup>[6]</sup>在准确地预测文件使用率、计算资源约束和

块的使用热度的基础上复制和传输数据,有效地缓和了热点,平均可以减少 20.2% 的工作完成时间.为了加强对中间数据的管理,研究者设计了中间数据存储系统,可以将工作执行时间可以减少约 45%<sup>[7]</sup>,通过并发,具有容错能力的最优化存储层存储中间数据可以有效把握中间数据的可用性,充分提高任务执行效率<sup>[8]</sup>.SplitCache<sup>[9]</sup>系统具有 OLAP 特征,它缓存计算节点的第一个输入流,以后需要时可重用,在很大程度上降低了数据传输冗余.Mantri<sup>[10]</sup>系统使用目标和资源感知技术减少任务完成时间不一带来的整体效率损失,通过重启延迟任务和保证最有价值任务的执行从而可以提高整体运行效率.

## 2.4 功能扩展研究

Hadoop 擅长处理海量的非结构化和半结构化数据,而不擅长处理关系型数据和大量小规模数据.为了扩大其使用范围,研究者将 Hadoop 与关系数据库、流处理等进行了结合研究.关系数据库擅长处理结构化数据, HadoopDB<sup>[11]</sup>将关系数据库与 MapReduce 进行了结合.它以 MapReduce 为主导,包括数据库连接器、数据加载器等组件,在数据加载、连接、查找、聚合等操作中均表现出较好的性能.有些应用需要定期分析大规模不断更新的数据流,例如股票交易系统,数据规模大,数据更新速度也很快.流处理可以应对每秒更新几百万次的海量数据,DEDUCE<sup>[12]</sup>延伸了 IBM 的流处理中间件,由 MapReduce 提供语言和运行支持并嵌入 MapReduce 工作队列.使用 PACT 和 Nephel<sup>[13]</sup>也可以将 MapReduce 与流处理结合,PACT 是 MapReduce 的一般化,Nephel 是一种可扩展的并行执行引擎,它使用专用协议保证了各项功能的效率,通过一种交流、同步和容错机制将 PACT 程序转换成 Nephel 数据流可以对不同类型的数据流实现最优化处理.另外,在数据挖掘、Web 排序、图像分析、模型试配等应用中需要多次重复执行 MapReduce,这种做法也被称为迭代应用.为了实现迭代应用,研究者开发了 Haloop<sup>[14]</sup>和 Twister<sup>[15]</sup>,Haloop 继承了 MapReduce 分布式计算模型,增加了新的索引算法,任务调度方面增加了环形控制和可变的缓存机制;Twister 在 MapReduce 基础上采用了专用的消息分发机制,通过广播和分散型通信进行数据传输.实验表明 Haloop 和 Twister 强化了 MapReduce 的运行能力,在迭代应用中表现出较好的性能.

### 3 Hadoop 应用研究

#### 3.1 云存储

Hadoop 分布式文件系统(HDFS)可以作为存储系统单独使用,相对传统的商业数据库系统 HDFS 提供了较好的扩展性和容错能力,并且建设成本低廉.使用 HDFS 弹性存储可以实现自动控制,灵活地进行存储空间的释放和分配,以适应快速变化的需求.例如在无线传感器网络中使用 Hadoop 可以快速建立存储系统,对海量数据进行存储和分析;数字广播电视系统使用 HDFS 处理数字视频资源,有利于实现双向服务和个性化定制等功能<sup>[16]</sup>.HDFS 在实际应用中也得到了一些改进,例如在虚拟化环境中优化虚拟机克隆机制可以节约能源消耗;DiskReduce<sup>[17]</sup>是 HDFS 的修改版本,是一种数据密集型可扩展 RAID,它将 HDFS 异步压缩的数据副本降低到 RAID 的冗余上限,增加了集群的存储能力.

#### 3.2 数据查询

在海量数据环境下进行数据查询工作对算法效率提出了较高的要求.传统的集群查询技术主要通过并行 DBMS 进行,由于并行 DBMS 缺乏协调容错机制,无法处理不可避免的软件和硬件错误.根据 MapReduce 设计的索引算法可以提供较高的容错性,实现较高的查询效率.根据 MapReduce 设计的索引算法目前主要有 Ivory 和 HIndex,前者利用 MapReduce 分布式建立索引,集中式提供服务,后者通过 HBase 分布式提供服务,但是索引集中建立.最近有研究者提出一种完全分布式索引算法,建立索引和提供服务都以分布式方式进行,实验表明这种索引算法索引速度快而且灵活性和健壮性都比较强<sup>[18]</sup>.采用 MapReduce 模型设计的 semi-join 算法可以克服经常遇到的数据不均衡问题,总体上降低计算量,提高磁盘读写效率;在 Map 和 Reduce 中间增加融合阶段可以简化关系数据处理,提高 join 效率<sup>[19]</sup>.MRShare<sup>[20]</sup>将批查询进行了融合减少了系统资源开销.SPIDER<sup>[21]</sup>系统专门面向 RDF 数据进行图片查询,它由图片加载器和查询处理器两部分组成,这两个模块都利用了 MapReduce 框架.

#### 3.3 数据分析

随着网络应用的快速增长,用户特征分析、点击流分析、社交网络分析和日志分析等大规模数据分析业务成为许多企业的重要工作,由于数据规模越来越

大,以 MapReduce 为基础的系统逐渐得到应用.很多互联网行业采用 Hadoop 搭建数据分析平台从超大规模用户行为数据中提取特征,并且应用到各种营销活动中.Cheetah<sup>[22]</sup>通过 Hadoop 使数据分析利用数据仓库中的数据,它是一个基于 MapReduce 定制的数据仓库,主要应用于在线广告,系统设计目标是实现类似 SQL 但是更加简明的查寻语言,以支持更多查寻特征,它在数据压缩和存取方法上进行了优化,处理原始数据速度可以达到 1GB/秒.其他行业在数据分析中也广泛在采用 Hadoop,美国宇航局的 DiscFinder 采用 Hadoop 集群作为底层的存储系统专门对超大规模天文学数据的存储和分析工作.

#### 3.4 数据挖掘

数据挖掘目的是在大规模数据中发现有价值的信息.随着数据规模越来越大传统方法难以应对多样化的业务需求,将 MapReduce 编程模型应用于数据挖掘可以解决这方面问题.Mahout 基于 Hadoop 已经实现了 Canopy 聚类、K-Means 聚类、Bayes 分类等数据挖掘算法.结合使用 MapReduce 和 DBMS 可以对广泛类型的数据集合进行分类挖掘,相对单纯使用 DBMS 具有很大程度的性能提升,并且降低了成本.很多互联网企业数据挖掘的对象是多媒体数据,使用 MapReduce 进行多媒体数据语义建模可以取得较快的学习速度,并且可以使过度适应风险变小<sup>[23]</sup>.

#### 3.5 关联广告

网络广告是互联网企业的基本盈利形式,多采用关联广告的形式放置在相关页面提高点击率.影响关联广告效果的关键是广告和页面内容的相关度,以雅虎等为代表的互联网公司建立了庞大的 Hadoop 集群专门为自身广告业务提供数据支持.采用页面计分函数、逻辑退化模型和标准余弦相似搭配等并行化分析工具能有效加快数据分析速度.通过仿真单一退火方法寻找全局最优参数,使用 NDCG 和 MAP 等信息检索方法可以提高广告相关性排列效果.采用 MapReduce 设计的 BT 框架具有很高的扩展能力,在数据不均衡时可以实现最佳的数据并行,任务并行和负载均衡,可以较好地支持行为定向广告<sup>[24]</sup>.

### 4 结论与展望

综合采用网格计算、并行计算和虚拟化技术的云计算将海量数据处理推进到一个新的时代.Hadoop 的开

源、高容错、跨平台等特点使其成为构建云平台的首选技术。目前国内的腾讯、百度、淘宝、阿里巴巴等互联网企业已经成功应用了 Hadoop, 教育、医疗、金融、制造业等行业采用 Hadoop 可以降低成本并提高效率。由于在国内应用时间不长, 以 Hadoop 为基础设计和开发实用高效的数据分析系统需要考虑具体生产环境中的众多因素。如何在保护已有投资的情况下研发出高效率的 Hadoop 数据处理平台需要广泛深入研究。

### 参考文献

- 1 DeWitt DJ, Paulson E, Robinson E, Naughton J. Clustera: An integrated computation and data management system. Proc. of the VLDB Endowment, 2008,1(1):28-41.
- 2 Isard M, Prabhakaran V, Currey J, Wieder U. Quincy: Fair scheduling for distributed computing clusters. Proc. of the ACM SIGOPS 22nd symposium on operating systems principles. New York: ACM, 2009: 261-276.
- 3 Park SM, Humphrey M. Predictable Time-Sharing for DryadLINQ Cluster. Proc. of the 7th International Conference on Autonomic Computing, New York: ACM, 175-184.
- 4 Henzinger TA, Singh V, Wies T, Zufferey D. Scheduling Large Jobs by Abstraction Refinement. Proc. of the Sixth Conference on Computer Systems. New York: ACM, 2011: 329-342.
- 5 Kwon YC, Balazinska M, Howe B, Rolia J. Skew-Resistant Parallel Processing of Feature-Extracting Scientific User-Defined Functions. Proc. of the 1st ACM Symposium on Cloud Computing. New York: ACM, 2010: 75-86.
- 6 Ananthanarayanan G, Agarwal S, Kandula S, Greenberg A. Scarlett: Coping with Skewed Content Popularity in Map Reduce Clusters. Proc. of the Sixth Conference on Computer Systems. New York: ACM, 2011: 287-300.
- 7 Ko SY, Hoque I, Cho B, Gupta I. Making Cloud Intermediate Data Fault-Tolerant. Proc. of the 1st ACM Symposium on Cloud Computing. New York: ACM, 2010: 181-192.
- 8 Leo S, Zanetti G. Pydoop: a Python MapReduce and HDFS API for Hadoop. Proc. of the 19th ACM International Symposium on High Performance Distributed Computing. New York: ACM, 2010.
- 9 Kim SG, Han H, Jung H, Eom H. Harnessing Input Redundancy in a MapReduce Framework. Proc. of the 2010 ACM Symposium on Applied Computing. New York: ACM, 2010: 362-366.
- 10 Ananthanarayanan G, Kandula S, Greenberg A, Stoica I. Reining in the Outliers in Map-Reduce Clusters using Mantri. Proc. of the 9th USENIX conference on Operating systems design and implementation. Berkeley: USENIX Association, 2010.
- 11 Gruska N, Martin P. Integrating MapReduce and RDBMSs. Proc. of the 2010 Conference of the Center for Advanced Studies on Collaborative Research. New York: ACM, 2010: 212-223.
- 12 Kumar V, Andrade H, Buğ ra Gedik, Wu KL. DEDUCE: At the Intersection of MapReduce and Stream Processing. Proc. of the 13th International Conference on Extending Database Technology. New York: ACM, 2010: 657-662.
- 13 Battré D, Ewen S, Hueske F, Kao O. Nephel/PACTs: A Programming Model and Execution Framework for Web-Scale Analytical Processing. Proc. of the 1st ACM Symposium on Cloud computing. New York: ACM, 2010: 119-130.
- 14 Bu YY, Howe B, Balazinska M, Erns MD. HaLoop: Efficient Iterative Data Processing on Large Clusters. Proc. of the VLDB Endowment, 2010,3(1-2):285-296.
- 15 Ekanayake J, Li H, Zhang BJ, Gunarathne T. Twister: A Runtime for Iterative MapReduce. Proc. of the 19th ACM International Symposium on High Performance Distributed Computing. New York: ACM, 2010: 810-818.
- 16 Lou JG, Fu Q, Wang Y, Li J. Mining Dependency in Distributed Systems through Unstructured Logs Analysis. ACM SIGOPS Operating Systems Review, 2010,44(1): 1461-1465.
- 17 Fan B, Tantisiriroj W, Xiao L, Gibson G. Disk Reduce: RAID for Data-Intensive Scalable Computing. Proc. of the 4th Annual Workshop on Petascale Data Storage. New York, ACM, 2009: 6-10.
- 18 Konstantinou I, Angelou E, Tsoumakos D, Koziris N. Distributed Indexing of Web Scale Datasets for the Cloud. Proc. of the 2010 Workshop on Massive Data Analytics on the Cloud. New York: ACM, 2010.
- 19 Yang HC, Dasdan A, Hsiao RL, Parker DS. Map-Reduce-

(下转第 28 页)

功能. 测试环节从后台管理上传发布资源到终端资源的访问, 测试了网络广播流媒体发布系统的功能.

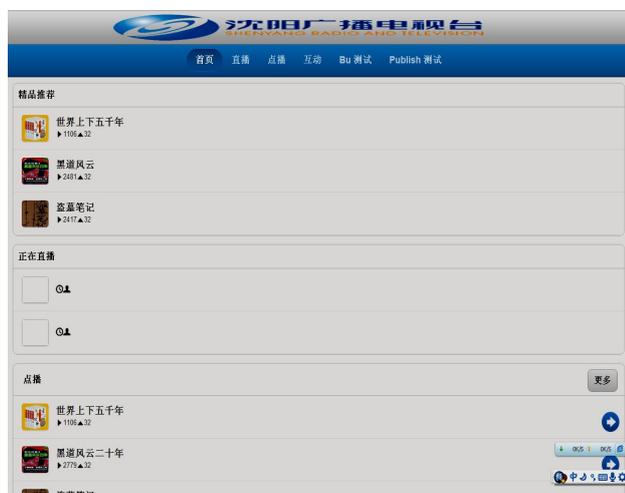


图4 Sencha Touch 业务终端截图

## 7 结论

本文针对网络广播设计开发了一个流媒体发布系统, 对媒体流的 Relay 和流媒体的发布两部分的设计与实现做了详细的描述. 最后通过测试验证了发布流程的可行性以及发布服务器的功能.

对流媒体发布系统来说, 需要进一步解决的问题: 发布服务器动态的请求源的问题、处理异构终端、不同的网络特性, 流格式等问题.

## 参考文献

- 1 唐静. 网络广播的优势及前瞻. 管理观察, 2009, (6): 203-204.
- 2 计叶. 基于 Darwin Streaming Server 快速流媒体中继实现. 中国科技论文在线.
- 3 夏斐. 实时视频监控系统中流媒体中继服务器的设计与实现[学位论文]. 北京: 北京邮电大学, 2011.
- 4 黄拔峰, 钟明, 杨传钧, 等. Darwin streaming Server 的研究与应用. 计算机工程, 2004, 30(19): 134-135.
- 5 杨碧华. 基于 Darwin 流媒体服务器的视频综合集成研究与应用[学位论文]. 广州: 暨南大学, 2011.
- 6 senchaTouch2.0. http://docs.sencha.com/touch/2-0/#[S].
- 7 袁进坤. 流媒体服务器中线程与缓存结构分析与研究[学位论文]. 南京: 南京邮电学院, 2004.
- 8 Apple Inc. QuickTime streaming server modules programming guide. http://images.apple.com/quicktime/pdf/QTSS\_Modules.pdf, 2005.
- 9 Apple Computer Inc. QTSS Modules Programming Guide, 2006.
- 10 曲柳莺. 流媒体传输协议的研究[学位论文]. 成都: 电子科技大学, 2005.
- 11 蒋铮. 基于达尔文流媒体服务器 TS 流推送模块的设计与实现[学位论文]. 成都: 电子科技大学, 2011.
- 12 张伟, 都志辉, 李三立. 达尔文流媒体服务器用户认证扩展与应用. 小型微型计算机系统, 2010, 31(4): 706-710.

(上接第5页)

- Merge: Simplified Relational Data Processing on Large Clusters. Proc. of the 2007 ACM SIGMOD International Conference on Management of Data. New York: ACM, 2007: 1029-1040.
- 20 Nykiel T, Potamias M, Mishra C, Kollis G. MRShare: Sharing Across Multiple Queries in Map-Reduce. Proc. of the VLDB Endowment, 2010, 3(1-2): 494-505.
- 21 Choi H, Son J, Cho YH, Sung MK. SPIDER: A System for Scalable, Parallel/Distributed Evaluation of large-scale RDF Data. Proc. of the 18th ACM conference on Information and Knowledge Management. New York: ACM, 2009: 2087-2088.
- 22 Chen ST. Cheetah: A high performance, custom data warehouse on top of MapReduce. Proc. of the VLDB Endowment, 2010, 3(1-2): 1459-1468.
- 23 Yan R, Fleury MO, Merler M, Natsev A. Large-Scale Multimedia Semantic Concept Modeling using Robust Subspace Bagging and MapReduce. Proc. of the First ACM workshop on Large-Scale Multimedia Retrieval and Mining. New York: ACM, 2009: 35-42.
- 24 Chen Y, Pavlov D, Canny JF. Large-Scale Behavioral Targeting. Proc. of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2009: 209-217.