# 基于云计算的汉字文化数字化平台的架构研究

杨 颐 张桂刚 王 健 黄卫星 苏海霞
(中国科学院自动化研究所 北京 100190)

**摘 要** 汉字是中华文明的核心元素,起到了记载中国历史和传承中华文化的重要作用。计算机科学为汉字文化数字化提供了重要的技术手段。在汉字文化的数字化信息日益增多的情况下,引入云计算和大数据技术进行数据存储、管理和分析,成为了汉字数字化的一个重要研究方向。汉字文化数字化系统便是在这种需求下研发的一套交互式汉字文化综合体验软件平台,其架构具有可扩展性、高可用性以及高安全性,并使用了数据预取、缓存机制等技术加快了数据的访问速度。同时,该系统提供了实时、非实时和半实时数据分析的功能,可以支撑汉字文化大数据的分析,更好地满足用户未来对汉字文化大数据的应用服务。该系统架构设计的有效性通过实验得到了验证。

**关键词** 汉字,数字化,云计算,大数据,Hadoop,架构
**中图法分类号** TP393.7 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.7.004

## Cloud Computing Architecture of Chinese Character Culture Digitization System

YANG Yi ZHANG Gui-gang WANG Jian HUANG Wei-xing SU Hai-xia
(Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China)

**Abstract** Chinese Character is a core element of Chinese civilization, which plays an important role in Chinese culture and history. Computer technology provides essential methods for Chinese character digitization. Chinese character culture digitization system (CCCDS) was developed for digitizing not only Chinese character but also Chinese culture around the characters. To deal with the rapidly increasing digitized Chinese character culture information, cloud computing and big data techniques were introduced as important means for data store, data management, and data analytics. The system provides an interactive user experience platform whose architecture is of scalable/scalability, high availability, and security. The system possesses real-time/history data analysis modules for the big data analysis in order to satisfy the requirements of applications and services based on Chinese character culture. The architecture of the system was validated by experiments.

**Keywords** Chinese character, Digitization, Cloud computing, Big data, Hadoop, Architecture

## 1 引言

汉字数字化技术涉及到不同的研究领域。汉字文化的数字化则包含更多的内容,除了汉字本身的数字化信息之外,还要通过研究汉字的内涵和外延,挖掘出汉字背后蕴藏的知识和所代表的文化,通过计算机技术进行存储、处理、分析后交互式地展现给用户,让用户对汉字文化有更深层次的理解,从而激发他们对汉字所代表的中华传统文化的兴趣。文中提出一种基于云计算和大数据技术的汉字文化数字化系统,该系统由两部分组成:汉字文化体验应用系统(简称体验系统)和云服务平台。体验系统是由各种交互式汉字文化体验应用子系统组成,比如汉字演绎体验系统、汉字构字法体验系统等;而云服务平台则为体验系统的各个子系统提供了软件运行环境、数据存储和管理服务,以及数据分析服务。

本文第2节介绍汉字数字化的研究进展;第3节和第4节分别介绍汉字文化数字化系统的总体设计思路和设计实践;最后,第5节使用实验验证了系统架构的有效性。

## 2 相关研究

汉字数字化的研究涉及到很多方面,最著名的便是王选主持研发的汉字激光照排技术,其将汉字数字化成点阵,极大地改进了汉字的印刷系统[1]。各种汉字输入法也是汉字数字化应用的具体产物[2]。同时,对汉字数字化的基础研究也大量涌现,完善了汉字在计算机中的基本存储格式,比如汉字编

到稿日期:2015-05-28 返修日期:2015-08-29 本文受2014年度中央文化产业发展专项资金(Y4T1011CA1),国家科技支撑计划重点项目(2015BAK25B04,2015BAK25B03)资助。

杨 颐(1978-),男,博士,助理研究员,主要研究方向为云计算、大数据、数据可视化,E-mail:yangyi@ia.ac.cn;张桂刚(1978-),男,博士后,副教授,主要研究方向为大数据、语意计算;王 健(1969-),男,博士后,副研究员,主要研究方向智能计算与控制、大数据;黄卫星(1974-),女,博士后,副教授,主要研究方向为文化传播、文化科技;苏海霞(1981-),女,博士,助理研究员,主要研究方向为控制科学与工程。

码技术[42]以及汉字编码之间的转换技术[43]。有了汉字编码技术作为基础，汉字才能通过计算机技术进行高效的处理。有的研究从汉字本身的结构方面进行数字化研究，比如根据汉字的偏旁部首，结合智能算法进行网络分析和聚类分析[44]。汉字的智能识别也是一个应用广泛的领域，比如手写识别研究[45]，以及对古籍上的汉字的数字化技术研究[4]。研究人员对出土文物上的汉字也做了数字化工作，比如对甲骨文的数字化技术研究[3]。面向汉字文化的数字化研究工作也有一定的进展，华东师范大学建立的中国文字数字化平台能够实现古今互译的功能，成为了汉字文化教学和传播的新方式。在云计算和大数据技术日益成熟的今天，汉字数字化研究也与大数据技术相结合，比如，研究人员利用大数据深度神经网络进行相似汉字的识别[46]，以及进行个性化手写识别的研究[47]。

## 3 平台架构总体设计思路

汉字文化数字化信息的数据量巨大，而且会随着相关研究的进展不断增加，数据格式多种多样，是典型的异构数据，比如汉字的释义和演绎历史是描述性文本数据，汉字的读音是音频数据，汉字的书写过程是视频格式。这些数据可以与用户行为数据相结合，通过大数据分析技术揭示出数据背后的知识和规律，让数据产生更大的价值，引发用户对汉字文化的兴趣。汉字文化数字化信息数据的这些特性符合大数据的4V特点，即Volume(大量)、Velocity(高速)、Variety(多样)、Value(价值)。因此，“云计算+大数据”的平台设计思路就成为了首选。Apache Hadoop[13-16]是目前最成熟的开源大数据平台，由HDFS和MapReduce构成。HDFS(Hadoop Distributed File System)[13,14]是一种可扩展、高可用的分布式文件系统，非常适合海量数据的存储。MapReduce[17]是基于Hadoop的高效的并行编程模型，与HDFS配合使用能满足大数据处理的高吞吐量的要求。近些年对MapReduce模型的研究使其功能更多样化，效率更高，比如MapReduce与语义技术相结合[50-54]。

体验系统由不同的子系统构成：汉字构字法系统、汉字演绎系统、汉字书写系统及汉字立体系统。子系统前端以Web网页的形式展现出来，采取了人机交互的设计原则以及可视化和虚拟现实的技术，用户可以使用浏览器对汉字衍生的文化要素进行交互式体验。子系统需要的数据访问大体可以分为几种：对描述性文本数据的访问、对图片文件的访问、对音频数据的访问、对视频数据的访问。体验系统所需的数据管理和计算则依赖于云服务平台提供的各种基础服务。

在设计云服务平台时，重点考虑了平台的高可用性(High Availability，HA)、可扩展性/可伸缩性(Scalability/Scalable)以及平台的安全性(Security)。

## 4 汉字文化数字化系统的架构设计

汉字文化数字化系统在设计上采取了云计算的三层架构的模式[5]，如图1所示，IaaS层和PaaS层构建了云服务平台，而SaaS层主要是用于体验系统的设计和运营。

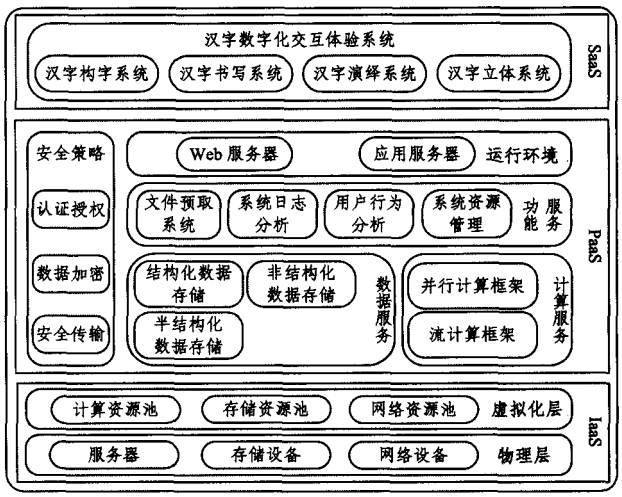



图1 汉字文化数字化系统的总体架构

### 4.1 IaaS层设计

IaaS(Infrastructure as a Service)层提供了汉字文化数字化系统的基础硬件设施。汉字文化数字化系统使用IBMFlex System作为物理主机，使用VMware的虚拟化技术[6]，构建了虚拟计算资源池、存储资源池和网络资源池，实现了弹性的资源利用和方便的资源管理。通过资源池中的虚拟机，构建了虚拟计算机集群，包括计算节点、存储节点、管理节点、安全保障节点等，节点数量可以根据性能的需求进行调整，达到灵活利用系统资源的目的。

### 4.2 PaaS层设计

PaaS(Platform as a Service)层提供了数据管理服务和各类型计算服务以及体验系统软件的运行环境，各模块间的关系如图2所示。

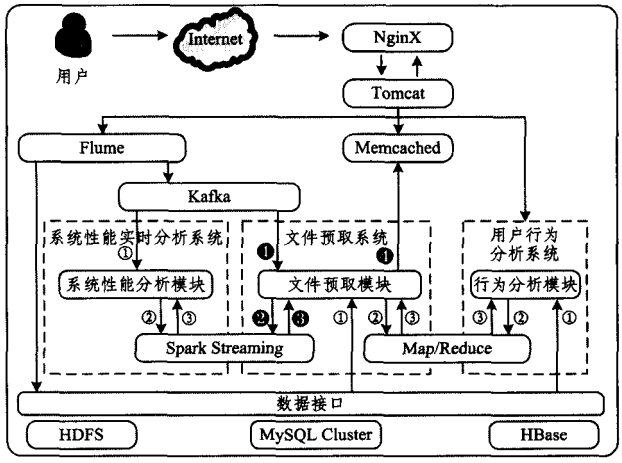



图2 汉字文化数字化系统的功能模块架构

#### 4.2.1 运行环境

体验系统是使用java开发的B/S结构Web软件，因此在PaaS层部署了Apache Tomcat[7,8]作为应用服务器，为体验系统的运行提供环境。考虑到Tomcat处理静态文件的效率不高，使用了动静分离技术，部署了NginX服务器[9,10]作为Web服务器，利用其强大的静态文件处理能力来处理HTML和jpg等数据；使用Tomcat作为Servlet容器，来处理JSP和servlet等动态文件。如果一个Tomcat服务器的性能达到上限，可以进行水平扩展，部署新的Tomcat服务器，并利用NginX的自动负载均衡的能力来自动调配Tomcat的计算资源。

4.2.2 数据管理

PaaS层除了提供体验系统的运行环境以外，也提供数据存储、管理以及计算等基础服务。汉字文化的数字化信息数据类型包含结构化数据和非结构化数据。结构化数据主要是汉字的一些描述性信息，包括释义、发展历史等。这些数据适合保存在关系型数据库中，因此使用 MySQL 来管理结构化数据。基于可扩展性和高可用性的考虑，使用了 MySQL Cluster 数据库集群技术来管理关系型数据库。如图 3 所示，MySQL Cluster[11]是一种分布式存储技术，在存储关系型数据方面很有优势[12]，数据存放在 NBD 存储服务器节点上，MySQL Cluster 使用无共享模式，把分布在不同数据节点的数据构建成一个内存数据库 NDB Cluster，并使用一个管理节点对这些数据节点进行管理和负载均衡。当一个数据节点出现崩溃时，数据会自动从其他节点复制，以恢复该节点的可用性。考虑到目前的数据量和可能的访问量，在 MySQL Cluster 中部署了 2 个 MySQL Server 节点和 4 个 NDB 存储节点。

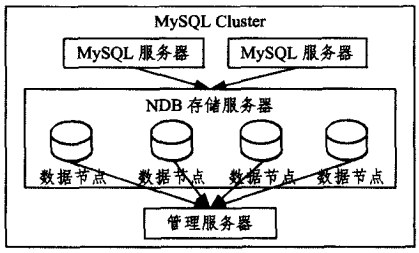



图 3 PaaS层的 MySQL 数据库集群

汉字文化数字化的数据还包括很多非结构化数据，比如音频、视频、图片。相比使用关系型数据库，这类文件更适合存放在文件系统中。比较了几种分布式文件系统，比如 Hadoop HDFS，Ceph[18,19]，MogileFS[20]等，选用了 Hadoop 框架的 HDFS 文件系统，除了 HDFS 本身具有更好的可扩展性和高可用性以外，考虑到后续功能中对大数据分析的需求，HDFS 与 Hadoop 的 MapReduce 编程模型具有优良的兼容性也是一个重要原因。然而，汉字文化数字化系统的数据包含大量的小文件，比如一张图片可能只有 5MB 容量。Hadoop HDFS 更适合存储 64MB 以上的文件，大量小文件会严重消耗 Hadoop NameNode 的内存资源，降低文件访问的效率。针对这一弊端，如图 2 所示，研发了文件预取系统，其可以通过机器学习的关联分析算法制定主动数据预取算法，预先将相关数据读取到缓存中，提高文件访问的命中率，这样可以缓解 HDFS 小文件访问的压力，细节介绍见 4.2.5 节。

4.2.3 缓存策略

如图 2 所示，在 PaaS 层部署了 Memcached 缓存服务器[21,22]，使用的缓存策略是基于访问时间的 LRU(Least Recently Used)算法。这样，应用程序访问数据时会先访问 Memcached 服务器，查找需要的数据，如果没有找到，再到 MySQL Cluster 访问结构化数据或者到 HDFS 文件系统中访问非结构化数据。Memcached 由于在内存中缓存数据和对象，因此比磁盘 I/O 要快很多。这样，就可以大幅提高数据访问的速度。

4.2.4 日志管理

汉字文化数字化系统的日志数据包含用户行为日志和系统资源使用日志，对日志数据的分析可以优化系统资源和性能，同时实现个性化汉字文化体验的推荐功能。

为了采集用户行为数据，使用了 Apache Flume[23]这种高可用、可扩展、分布式的日志采集系统。Flume 扩展性很强，能够像积木一样组合搭建，进行并行处理。Flume 的 agent 对用户行为数据进行监控，并将新数据送往 HBase[24-27]数据库长期储存起来，同时送往 Apache Kafka[35,36]用于后续分析。Flume 将消息按照类型分别发布到 Kafka 的不同 Topic 中，消息可以短时间保存在其中。系统的数据分析系统作为 Kafka 的数据消费者，会主动按需要读取不同 Topic 中的数据来进行数据分析。分析系统中的不同模块的数据需求会有交集，使用消息订阅模式可以实现多个模块同时使用同一消息数据的需求。

4.2.5 数据分析系统

数据分析是个大主题，在汉字文化数字化系统中主要用于系统资源监控、用户行为分析和文件预取策略计算。云服务平台提供了 3 种分析模式：实时统计分析、非实时数据分析和半实时数据分析。实时统计分析用于系统资源的监控，非实时数据分析面向用户行为分析，半实时数据分析用于文件预取系统，各模块的内部流程如图 2 所示。

系统资源实时性能分析系统监测着各种系统资源的使用情况，比如平均响应时间、吞吐量等指标，这需要实时地对采集到的日志数据进行统计分析。系统运维人员通过实时分析可以及时发现系统的问题，采取相应的措施，进行系统优化。为了满足实时分析的需求，PaaS 层部署了 Apache Spark 计算框架[28,30]，利用其提供的 Spark Streaming 流计算框架来实现实时统计分析。Spark Streaming 具有高可用和可扩展性的特点，可以将收到的实时连续数据离散化为 RDD 数据块，然后对相同时间段内的 RDD 数据进行并行计算处理，由于计算过程在内存中进行，因此 Spark Streaming 的计算实时性可以达到秒级。实现流程如图 2 所示，系统性能实时分析模块从 Kafka 中读取所需的日志数据(①)，利用 Spark Streaming 进行实时统计分析(②)，然后返回结果(③)。分析结果以可视化的方式展现在 Web 页面上。

用户行为分析是指根据用户对体验系统的访问数据(比如登录地点、登录时间、页面访问次数、页面停留时间、汉字的查询频率等信息)，来分析用户的行为模式。用户行为分析是对海量历史数据的分析，没有很强的实时性需求，更多的是注重分析模块的数据吞吐能力，因此选用 Hadoop MapReduce 作为分析算法的编程模型。实现流程如图 2 所示，用户行为分析系统的分析模块首先从数据库中提取数据(①)，然后利用 Hadoop MapReduce 编程模型结合数据挖掘的协同过滤算法进行海量日志数据计算分析(②)，得出分析结果，即需要推荐的内容(③)。之后，用户行为分析系统可以将用户感兴趣的汉字文化的内容推荐给用户。

文件预取指的是利用算法预测应用系统需要访问的数据，并将其提前读取到缓存中，以加快数据访问的速度，优化系统性能。文件预取分析系统是一种结合了实时分析和非实时历史数据分析的半实时数据分析系统，通过预取分析模块来计算预取策略，得知应该提前预取的文件。整个流程分为两部分，一部分进行非实时分析，另一部分进行实时分析，两部分运行相对独立。一方面，对存入 HBase 的日志数据利用关联分析算法进行分析，找出被访问文件的关联性；另一方面，利用流技术进行实时统计，得出文件近期访问的频率。然

后结合频率和文件访问的关联性，进行加权，最后计算出最有可能被访问到的文件。实现流程如图 2 所示，文件预取系统通过数据接口从 HBase 系统中读取日志数据(①)，然后使用 Hadoop MapReduce 模型进行关联性分析(②)，并得到关联分析的结果数据。关联分析针对历史数据进行分析，每 2 小时会执行一次，并更新结果数据(③)。另一方面，文件预取模块从 Kafka 中读取最新的日志数据(❶)，然后利用 Spark Streaming 实时统计计算(❷)，之后，文件预取算法会将当前的统计结果和关联分析出来的数据进行加权计算并得到结果，即关于文件访问可能性的分值(❸)，最后按照分值大小将最有可能被访问的文件预先读取到 Memcached 缓存系统(❹)。这样就可以提高缓存的命中率，加快文件访问的速度。

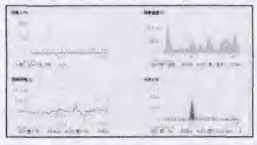
#### 4.2.6 系统资源管理

在 PaaS 层设计实现了一个系统资源管理模块，来监测汉字文化数字化系统的资源利用和管理数据，如图 4 所示，其主要包括 3 个部分。

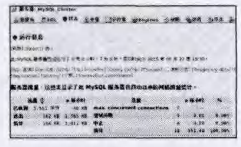
• 系统资源利用情况的监测。使用 Cloudera CDH[48] 开源平台管理 Hadoop 系统。因此，将在图形界面上嵌入 Cloudera Manager 的 GUI 作为大数据的可视化监控界面。除了可以对 CPU 利用率、网络 I/O、磁盘 I/O 等系统资源的使用情况进行监测外，还可以可视化地对 Hadoop 的组件以及虚拟机进行管理。

• MySQL Cluster 集群的监测和管理。在系统管理模块的 GUI 界面中嵌入 phpMyAdmin[49] 作为 MySQL Cluster 的管理界面，可以监控到 MySQL 服务器的运行状况，并对 MySQL 所储存的数据进行管理。

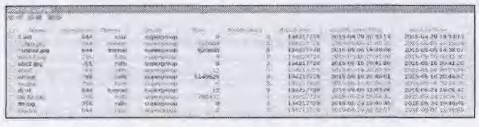
• HDFS 文件系统的管理。HDFS 中存储着重要的业务数据和日志，为了能够安全地管理这些数据，开发了针对 HDFS 文件管理的模块，对文件进行安全的增删查改操作。安全文件操作体现在两个方面，一方面是对管理员的严格的授权管理，如果需要删除或者更新文件，必须通过两个超级管理员的验证才可以被允许；另一方面，需要删除或者更新的文件会先被备份到一台数据备份服务器上，然后再从 HDFS 上删除掉，安全审计人员会定期检查文件操作，如果出现未经允许的操作，就会从备份服务器上对数据进行恢复。



(a) 系统资源监测



(b) MySQL Cluster 监控



(c) Hadoop HDFS 文件系统管理

图 4 汉字文化数字化资源管理系统

### 4.3 SaaS 层设计

SaaS(Software as a Service)[33,34] 层主要集中在体验系统的设计。SaaS 软件系统的重点就是多租户模式和按需提供服务。多租户模式[39,40] 是指租户可以共用一个或一组程序实例，如图 5 所示。在实现方面，遵从 SaaS 成熟度模型[41] 的第四级的思想，即可扩展、可配置与多租户高效率，设计了一个实例池，其默认包含了体验系统程序的 3 个实例，各部署在一台 Tomcat 服务器上，如图 5 所示。所有租户共享这个实例池，实例池根据实际运营的性能指标，比如数据吞吐量，来调整实例池中的计算节点的数量，并进行负载均衡。体验系统通过配置文件来实现每个租户专有的 UI 和功能。这样就实现了系统资源和功能两个层面上的按需提供服务。

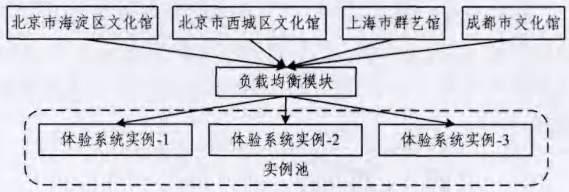



图 5 汉字文化数字化系统多租户设计

### 4.4 安全策略设计

安全问题是计算机系统的重点研究方向，对于大数据平台尤为重要。我们为汉字文化数字化系统设计了多种安全策略来保障数据和用户隐私的安全。

• 虚拟主机之间对文件的安全访问，使用了 Kerberos＋SSL 的方案。Kerberos 协议[31,32] 用于机器级别的授权认证，没有经过认证的主机无法访问数据，以防止有人通过未经认证的主机恶意访问被保护的数据。SSL(Secure Sockets Layer) 安全协议[37,38] 使用密钥方式加密传输中的数据，保证数据传输过程中不会被窃取。同时 SSL 还会对传送的数据进行完整性验证，保证数据传输过程中没有损失。

• 云服务平台采用了基于角色的访问控制(Role-Based Access Control)，对不同的角色分配不同的访问权限，根据用户所属的角色，就可以限制用户能够访问的数据范围。

• 外界用户是通过浏览器来访问体验系统的，如果将应用服务器 Tomcat 的 IP 直接暴露出来，其就会很容易被攻击。因此，如图 2 所示，部署了 NginX 服务器作为反向代理服务器，起到网络防火墙的作用，把 Tomcat 服务器对外接用户隐藏起来。

## 5 汉字文化数字化系统架构的性能分析

### 5.1 实验说明

对云服务平台架构设计进行了实验，并对结果进行分析。对于云平台的实验重点在于云平台整体的有效性和扩展性，而不关注架构所采用的单项技术，因为这些单项技术的优点已经被很多研究验证过，比如 MapReduce 的并行计算优势、HDFS 的分布式存储能力等。

云服务平台由于是基于分布式架构的系统，因此在扩展性方面会比传统系统表现得更加优异，但是在访问速度方面不会有优势。因此，本实验需要验证的假想是：相对于传统架构系统而言，云服务平台能够在保证性能的情况下提供更好的扩展能力。实验的重点有 2 个：云服务平台的性能及云服务平台的可扩展性。

### 5.2 云服务平台架构的整体性能分析

将云服务平台整体的性能与传统架构系统做性能比较。实验的指标是发送请求的并发量、系统的平均响应时间和吞吐量。请求并发数量是指同一时刻向服务器发送请求的数

量。响应时间(Response Time)是从用户发出请求到得到返回结果的时间差,单位是毫秒(ms)。时间差不包含客户端浏览器渲染的时间,因为渲染时间受到客户端配置的影响极大。平均响应时间是在长时间发送大量请求的情况下统计出的均值。吞吐量(throughput)是另外一个性能评测的重要指标,使用每秒处理请求的数量(requests/second)来评估。

5.2.1 实验设计

比较分布式的云服务平台与传统的非集群化的 B/S 系统的性能,如图 6 所示。用户通过网络访问部署在 Tomcat 上的软件系统,Tomcat 服务器与 MySQL 和 Linux 文件系统连接。

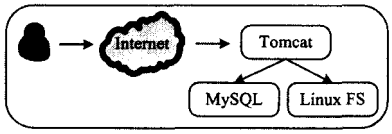



图 6 传统系统架构

为了保持实验环境的一致性,两个实验系统部署在相同的云环境上。传统架构系统由 3 台虚拟机组成,分别是 Tomcat 服务器、MySQL 服务器、Linux 文件服务器。云平台架构包含 1 台 NginX 服务器、1 台 Tomcat 服务器、2 台 MySQL Cluster、3 台 Hadoop。实验环境的基本配置如表 1 所列。两个实验系统使用的软件均采用默认配置,未进行专门优化。

表 1 实验环境

| 系统 | 虚拟机配置 | 软件环境 |
|---|---|---|
| 云服务平台 | CPU:4 核 2.0GB | CentOS 6.6 |
| | 内存:8GB,Hadoop | Nginx 1.6.2 |
| | 主节点:16GB | Tomcat 7 |
| | 硬盘:100GB | MySQL Cluster 7.4.6 |
| | 网络:1000M | Hadoop 2.5 |
| 传统架构系统 | CPU:4 核 2.0GB | CentOS 6.6 |
| | 内存:8GB | Tomcat 7 |
| | 硬盘:100GB | MySQL 5.6.23 |
| | 网络:1000M | |

5.2.2 实验工具及方法

使用 Apache JMeter[55] 作为实验的工具,JMeter 能够使用多线程向服务器发送请求,常用于软件系统负载测试。使用 JMeter 以线性增长的方式分别向传统架构系统和云服务平台发送 200 个请求,并统计出吞吐量和平均响应时间。请求的内容包括主页访问、访问 MySQL 数据库,以及磁盘文件访问的 Http 请求。

5.2.3 实验结果及分析

实验结果如表 2 所列。

表 2 实验结果

| 并发请求数 | 平均响应时间-传统架构 | 吞吐量-传统架构 | 平均响应时间-云平台 | 吞吐量-云平台 |
|---|---|---|---|---|
| 20 | 100 | 185 | 121 | 163 |
| 40 | 130 | 295 | 152 | 255 |
| 60 | 155 | 361 | 170 | 348 |
| 80 | 162 | 477 | 179 | 436 |
| 100 | 175 | 564 | 185 | 532 |
| 120 | 181 | 653 | 193 | 615 |
| 140 | 289 | 472 | 312 | 439 |
| 160 | 368 | 440 | 394 | 398 |
| 180 | 420 | 418 | 480 | 366 |
| 200 | 485 | 412 | 520 | 375 |

图 7 显示了平均响应时间随着并发请求量增大所产生的变化。实验结果表明,云服务平台的平均响应时间高于传统架构系统大约 15%,原因是云平台的分布式架构上有很多节点间的消息传递造成延时。对于 Web 应用的后端访问速度,并发数量在 120 以内时,云平台的平均响应时间的数值是可以接受的。

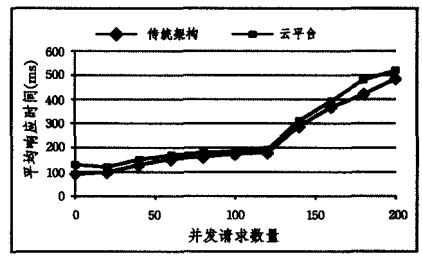



图 7 平均响应时间的实验结果

图 8 显示了系统吞吐量随着并发量增大所产生的变化。实验结果表明,云服务平台的吞吐量略微低于传统架构系统,这同样可以解释为分布式系统的特性所造成的轻微性能下降。从总的发展趋势来看,两个系统的吞吐量都是在并发请求量到达 120 时达到峰值,之后出现了吞吐量大幅下降的情况。从吞吐量的数值来看,云平台的每秒处理请求的能力是完全可以接受的。

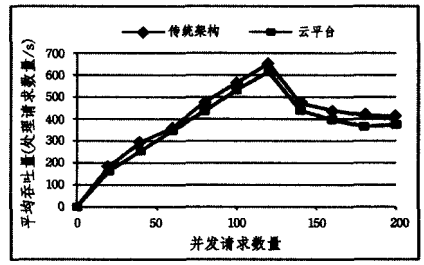



图 8 吞吐量的实验结果

从实验数据分析的结果可以得出结论:云服务平台的系统性能与传统架构系统的性能基本一致,符合 Web 应用系统的性能要求。

## 5.3 云服务平台扩展性分析

从平均响应时间变化的趋势来分析,并发量在 120 的时候出现拐点,之后大幅增长,这说明系统性能出现了瓶颈,性能开始下降,同样的情况出现在吞吐量实验上。综合分析,云平台系统在并发量超过 120 的时候,处理速度降低,处理时间变长,由此可以得出结论:云平台系统在未进行扩展的情况下能够处理的并发请求数量在 120 左右,这也是其基本架构的处理能力瓶颈。在系统出现瓶颈的情况下,对云平台进行水平扩展,以增强处理能力。因此,本实验的目的在于证明云服务平台的架构有能力通过系统的水平扩展来增大并发处理能力。

5.3.1 实验设计

由于云平台由不同的技术模块构成,因此需要扩展的模块也需要根据实际情况来判断,这就对实验的设计提出了很高的要求。为了降低实验的复杂程度,采用了简单方式进行实验来验证云平台有一定的扩展能力,但不进行更加详细的定量分析。进行了 4 个实验,每个实验的架构都会在前一个

实验的基础上进行了水平扩展，记录并分析出每个实验的吞吐量瓶颈。实验的架构配置中，所包含的模块节点的数量如表3所列。

表3 云平台扩展性实验：模块节点数量配置

| 模块 | 实验1 | 实验2 | 实验3 | 实验4 |
|---|---|---|---|---|
| NginX | 1 | 1 | 1 | 1 |
| Tomcat | 1 | **2** | **4** | 4 |
| MySQL Server | 2 | 2 | 2 | **4** |
| Hadoop Node | 3 | 3 | 3 | **6** |

实验1设置了1台NginX、1台Tomcat、2台MySQL Cluster和3台Hadoop节点；实验2扩展了Tomcat；实验3进一步扩展了Tomcat；实验4扩展了MySQL Cluster和Hadoop的节点。

5.3.2 实验结果及分析

如图9所示，结果显示随着云服务平台的扩展，其吞吐量也会相应增长，系统的处理能力得到加强。实验结果证明了云服务平台的架构能够通过水平扩展加强其并发处理的能力。

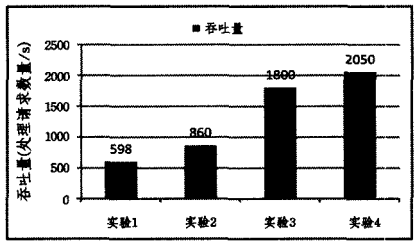



图9 云平台扩展性实验结果

## 5.4 实验分析总结

通过以上的系统性能实验和系统扩展性实验，可以得出结论：云服务平台的架构在性能完全符合应用要求的情况下，与传统架构的系统性能基本一致，但拥有更好的扩展性，能够根据实际需要增强系统性能。该结论证明了此前的实验猜想。通过实验，云服务平台架构设计的有效性得到了验证。

**结束语** 本文介绍了基于云计算技术的汉字文化数字化系统的架构设计，包括存储模型、计算模型、安全策略、缓存策略的设计，以及数据分析系统的设计。该系统充分利用了云计算的虚拟计算机集群技术、实时流计算技术和MapReduce计算模型，实现了系统资源的弹性利用、高可用性、平台安全、数据分析以及对系统的优化。在今后的工作中，架构研究的重点将会是加强架构设计的SOA化。随着各种新体验系统的研发，大量可复用的功能可以抽象成服务，一个完善的SOA云架构将会大幅度提高云服务平台的整体效率，让用户感受到更好的汉字文化数字化体验。

## 参考文献


[1] Wang Xuan. A brief introduction to the computerized Chinese Character Editing and laser type setting system[J]. Chinese Journal of Computers, 1981(2):83-89(in Chinese)
王选. 计算机-激光汉字编辑排版系统简介[J]. 计算机学报, 1981(2):83-89

[2] Yu Shi-wen. The Application of Grammatical Analysis Technique in Chinese Input[J]. Journal of Chinese information processing, 1988, 2(3):20-26(in Chinese)
俞士汶. 中文输入中语法分析技术的应用[J]. 中文信息学报, 1988, 2(3):20-26

[3] Ma Xiao-hu, Yang Yi-ming, Huang Wen-fan, et al. Research on the Technology of the Automatic Generation "Jiaguwen" Outline Font and Building of Universal Jiaguwen Font[J]. Applied Linguistics, 2004(3):105-111(in Chinese)
马小虎，杨亦鸣，黄文帆，等. 甲骨文轮廓字形生成技术研究与通用甲骨文字库的建设[J]. 语言文字应用, 2004(3):105-111

[4] Chen Bin-ren. An Attempt to Digitize Ancient Rare Books[J]. New Technology of Library and Information Service, 1998(1):22-25(in Chinese)
陈秉仁. 古籍善本数字化的尝试[J]. 现代图书情报技术, 1998(1):22-25

[5] National Institute of Standards and Technology[S]. 2011

[6] Muller A l, Wilson S. Virtualization with Vmware EsxServer [M]. Syngress Publishing, 2005

[7] Apache Tomcat[OL]. http://tomcat.apache.org

[8] Brittain J, Darwin I F. Tomcat: The Definitive Guide(2nd Edition)[M]. O'Reilly Media, 2007

[9] Nginx[OL]. http://nginx.org

[10] Reese W. Nginx: the high-performance web server and reverse proxy [J]. Linux Journal, 2008, 2008(173)

[11] MySQL. MySQL AB[OL]. http://www.mysql.com

[12] Bunch C, Chohan N, Krintz C, et al. An Evaluation of Distributed Datastores Using the AppScale Cloud Platform[C]// Proceedings of the 2010 IEEE 3rd International Conference on Cloud Computing, 2010(CLOUD'10). Washington, DC, USA: IEEE Computer Society, 2010:305-312

[13] Hadoop[OL]. http://hadoop.apache.org

[14] Shvachko K, Kuang Hai-rong, Radia S, et al. The Hadoop Distributed File System[C]// 2010 IEEE 26th Symposium Mass Storage Systems and Technologies (MSST), 2010. Incline Village, NV: IEEE, 2010:1-10

[15] Venner J. Pro Hadoop [M]. Apress, 2009

[16] White T. Hadoop: The Definitive Guide[M]. O'Reilly Media, Yahoo Press, 2009

[17] Dean J, Ghemawat S. MapReduce: Simplified Data Processing on Large Clusters[C]// Sixth Symposium on Operating System Design and Implementation 2004 (OSDI'04). New York, NY, USA: ACM, 2008:107-113

[18] Ceph[OL]. http://ceph.com

[19] Weil S A, Brandt S A, Miller E L, et al. Ceph: A Scalable, High-Performance Distributed File System[C]// Proceedings of the 7th Symposium on Operating Systems Design and Implementation (OSDI). Seattle, WA, Berkeley, CA, USA: USENIX Association, 2006:307-320

[20] Danga Interactive. MogileFS[OL]. http://www.danga.com/mogilefs

[21] Mencached[OL]. http://memcached.org

[22] Fitzpatrick B. Distributed caching with memcached [J]. Linux Journal, 2004, 2004(124):72-76

[23] Apache Flume[OL]. https://flume.apache.org

[24] Apache HBase[OL]. http://hbase.apache.org

[25] Chang F, Dean J, Ghemawat S. Bigtable: a distributed storage system for structured data[J]. ACM Transactions on Computer

Systems,2008,26(2):205-218

[26] Khetrapal A, Ganesh V. HBase and Hypertable for large scale distributed storage, systems: A Performance evaluation, for Open Source BigTable Implementations [EB/OL]. http://www. ankurkhetrapal. com/downloads/Hypertable. HBaseEval2. pdf

[27] Carstoiu D, Cernian A, Olteanu A. Hadoop Hbase-0. 20. 2 performance evaluation[C]// 2010 4th International Conference on New Trends in Information Science and Service Science (NISS), 2010. IEEE, 2010:84-87

[28] Apache Spark[OL]. http://spark. apache. org

[29] Zaharia M, Chowdhury M, Franklin M J, et al. Spark: Cluster Computing with Working Sets[C]// Proceedings of the 2nd USENIX conference on Hot topics in cloud computing, 2010 (HotCloud'10). Berkeley, CA, USA: USENIX Association, 2010:1765-1773

[30] Zaharia M, Chowdhury M, Das T, et al. Spark: Resilient distributed datasets:a fault-tolerant abstraction for in-memory cluster computing[C]// Resilient Distributed Datasets: a fault-tolerant Abstraction for in-memory Cluster Computing, 2012. Berkeley, CA, USA: USENIX Association, 2012:141-146

[31] Kerberos[OL]. http://www. kerberos. org

[32] Kohl J, Neuman C. The Kerberos Network Authentication Service (V5) [R]. United States: RFC Editor, 1993

[33] Kwok T, Nguyen T, Lam L. A software as a service with multi-tenancy support for an electronic contract management application[C]// Proceeding Int. Conf. on Services Computing (SCC), 2008. Washington, DC, USA: IEEE Computer Society, 2008: 179-186

[34] Wang Zhi-hu, Guo Chang-jie, Gao Bo, et al. A study and performance evaluation of the multi-tenant data tier design patterns for service oriented computing[C]// Proceeding of the International Conference on e-Business Engineering (ICEBE), 2008. Washington DC, USA: IEEE Computer Society, 2008:94-101

[35] Apache Kafka[OL]. http://kafka. apache. org

[36] Kreps J, Narkhede N, Rao J. Kafka: A distributed messaging system for log processing[C]// Proceedings of 6th International Workshop on Networking Meets Databases (NetDB). Athens, Greece: ACM, 2011

[37] Karlton F P, Kocher P. The SSL3. 0 Protocol [R]. Netscape Communications Corp, 1996

[38] Dierks T, Allen C. The TLS Protocol Version 1. 0[R]. IE TF RFC2246, January 1999

[39] Kaplan M J. SaaS Survey Shows New Model Becoming Mainstream[J]. Cutter Consortium Executive Update, 2005, 6(22): 1-5

[40] Chong F, Carraro G, Wolter R. Multi-Tenant Data Architecture [EB/OL]. (2006). https://msdn. microsft. com/en-us/library/aa479086. aspx

[41] Chong F, Carraro G. Architecture Strategies for Catching the Long Tail[EB/OL]. (2006). https://msdn. microsoft. com/en-us/library/aa479069. aspx

[42] Tu Jian-guo. On the Coding and Composing of Chinese Character[J]. Library, 2002(1):60(in Chinese)
涂建国. 汉字编码和汉字排检法[J]. 图书馆, 2002(1):60

[43] Ni Xiao-jun. A High-Performance Unicode/GB Transcoding Algorithm[J]. Computer Technology and Development, 2009, 19(9):21-24(in Chinese)
倪晓军. 高效 Unicode/GB 编码转换算法的设计和实现[J]. 计算机技术与发展, 2009, 19(9):21-24

[44] Han Ying, Li Jian-yu, Huang Xiang-lin, et al. The complex networks of the parts word in Chinese structure[J]. Journal of Harbin Engineering University, 2006, 27(z1):580-583(in Chinese)
韩莹, 李健瑜, 黄祥林, 等. 构成汉字偏旁字符的复杂网络[J]. 哈尔滨工程大学学报, 2006, 27(z1):580-583

[45] Lu Hao-ru, Yang Yuan-yuan. A General Discussion For Handprinted Chinese Character Recognition[J]. Computer Applications and Software, 1994(2):1-8(in Chinese)
路浩如, 杨源远. 手写体汉字识别问题综论[J]. 计算机应用与软件, 1994(2):1-8

[46] Yang Zhao, Tao Da-peng, Zhang Shu-ye, et al. Similar handwritten Chinese character recognition based on deep neural networks with big data[J]. Journal on Communications, 2014, 35(9):184-189(in Chinese)
杨钊, 陶大鹏, 张树业, 等. 大数据下的基于深度神经网的相似汉字识别[J]. 通信学报, 2014, 35(9):184-189

[47] Zhou Gui-bin. Personalized Handwritten Chinese Character Recognition System Based on Cloud Computing[D]. Guangzhou: South China University of Technology, 2012(in Chinese)
周贵斌. 基于云计算平台的个性化手写识别系统的研究[D]. 广州: 华南理工大学, 2012

[48] Cloudera[OL]. http://www. cloudera. com

[49] phpMyAdmin[OL]. http://www. phpmyadmin. net

[50] Zhang Gui-gang, Li Chao. A Semantic++ MapReduce Parallel Programming Model [J]. International Journal of Semantic Computing, 2014, 8(3):1-21

[51] Zhang Gui-gang, Li Chao, Zhang Yong, et al. MapReduce++: Efficient Processing of MapReduce Jobs in the Cloud [J]. Journal of Computational Information Systems, 2012, 8(14):5757-5764

[52] Zhang Gui-gang, Wang Jian, et al. A Semantic++ MapReduce-A Preliminary Report[C]// 2014 IEEE International Conference on Semantic Computing, 2014. Newport Beach, CA, USA, 2014: 330-336

[53] Zhang Gui-gang, Li Chao, Zhang Yong, et al. Massive Data Query Optimization on Large Clusters[J]. Journal of Computational Information Systems, 2012, 8(8):1391-1398

[54] Zhang Gui-gang, Li Chao, Zhang Yong, et al. An Efficient Massive Data Processing Model in the Cloud—A Preliminary Report [C]// Proceedings 7th ChinaGrid Annual Conference, 2012. IEEE Computer Society Press, 2012:148-155

[55] Apache JMeter[OL]. http://jmeter. apache. org