

面向新媒体广播的云 CDN 系统^①

于 波², 张寒冰^{1,2}

¹(中国科学院大学, 北京 100049)

²(中国科学院 沈阳计算技术研究所, 沈阳 110168)

摘要: 互联网用户的快速增长使得流媒体访问需求不断增加, 庞大的流媒体访问需求和有限的服务器分发能力, 造成了系统响应慢、用户体验下降等问题. 针对以上问题, 结合新媒体广播项目中流媒体分发的实际应用, 提出了一种基于云计算的流媒体分发系统, 主要利用云计算的分布式处理和 CDN(Content Distributed Network) 分发系统来使用户能够就近访问所需资源, 在保证流媒体分发质量的同时提高分发速度, 降低企业成本改善用户访问体验. 本文主要阐述了面向新媒体广播的云 CDN 系统的总体框架和主要功能, 具体介绍了借助于 Hadoop 实现的客户端测速技术以及基于用户访问行为的缓存策略.

关键词: 云计算; 流媒体分发; CDN; 负载均衡; 缓存策略

CDN System Based on Cloud Computing for New Media Broadcasting

YU Bo¹, ZHANG Han-Bing^{1,2}

¹(University of Chinese Academy of Sciences, Beijing 100049, China)

²(Shenyang Institute of Computing Technology, Chinese Academy of Sciences, Shenyang 110168, China)

Abstract: The rapid growth of the number of Internet users makes the visiting requirements of streaming media continue to increase, the huge demand for streaming media access and the limited ability of servers to distribute streaming media cause some problems, such as slow system response and the decrease of user experience. To solve these problems and apply the solution to the New Media Broadcasting the CDN system based on cloud computing is proposed, it mainly uses cloud computing and CDN to enable the users to access the nearest required resource, to ensure the quality while it increases the distribution speed of streaming media and reduce costs to improve user experience. This paper mainly describes the structure and major functions of the CDN system based on cloud computing for New Media Broadcasting, specifically introduces the client speed measurement technology by means of Hadoop and a caching technology based on user behavior.

Key words: cloud computing; distribution of streaming media; CDN; load balance; caching techniques

随着互联网的迅猛发展, 传统媒体逐步向网络媒体、手机媒体等新媒体转换, 新媒体所催生的实时广播、视频点播等流媒体业务在互联网业务中所占的份额也越来越大, 日益庞大的数据量以及流媒体本身具有的持续时间长、处理终端多样化、对传输质量以及处理速度要求高等特点使得流媒体处理面临着服务器响应慢、网络拥堵、资源浪费、用户体验下降等问题^[1]. 现有的流媒体分发技术中比较常用的是基于 CDN(Content Distributed Network) 的分发策略. CDN

的主要思想是尽量使得视频服务靠近终端的同时利用终端资源来解决网络堵塞问题, 但是, CDN 资源采用静态部署方式, 会导致资源部署不足或浪费^[2,3]. 云计算的弹性计算特性有助于解决这些问题, 运用云平台的大型资源池可以减少开支的同时避免资源的浪费, 提高数据处理速度和质量^[4,5]. 中国科学院沈阳计算技术研究所承担的新媒体广播项目涉及到在线广播和音视频访问等流媒体业务, 正面临着此类问题. 为此设计了一种基于云计算的适用于新媒体广播流媒体分发

① 收稿时间:2015-11-23;收到修改稿时间:2015-12-21 [doi:10.15888/j.cnki.csa.005247]

的 CDN 系统, 来达到加快流媒体分发速度, 提高并发访问处理能力, 提升用户访问体验的效果。

1 系统总体设计

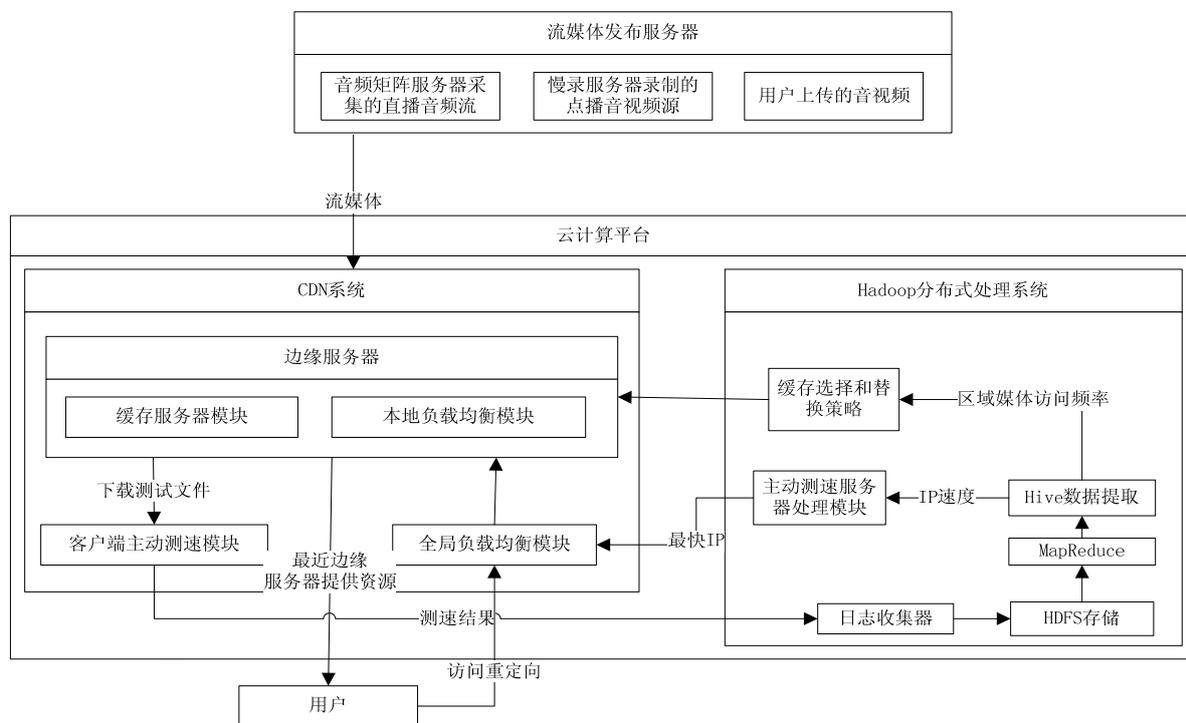


图 1 系统架构

针对新媒体广播所设计的流媒体分发系统结构如图 1 所示, 主要是对新媒体发布服务器中的音视频进行分发处理。当用户访问媒体资源时分发过程如下:

① 用户通过域名地址访问资源, 域名解析时授权的 DNS 将域名解析为 GSLB 全局负载均衡服务器的 IP 地址返回给用户。

② 用户向 GSLB 发起请求, GSLB 根据客户端主动测速结果将用户请求重定向到离用户最近的, 能够高效提供服务的边缘节点。

③ 边缘节点中的本地负载均衡服务器根据连接数最小的调度策略转发到适当的缓存服务器, 检查是否存在相应资源、资源是否过期, 若本地并无缓存文件或者资源已经过期, 则继续向上级节点请求内容。

④ 边缘节点向上级节点发出文件请求, 上级节点查找是否命中, 若无则请求源站, 源站将所需文件返回给用户, 边缘节点存储该文件, 下次访问时可直接提供该资源。

2 系统主要模块设计

2.1 OpenStack 云平台

OpenStack 是用来搭建与管理公共及私有云的软件, 其中的 OpenStack Compute 模块提供了一个工具来组织云, 可以管理网络, 开启和执行虚拟机实例, 通过用户和项目来控制对云的使用。利用 OpenStack 搭建云平台, 认证之后可以使用浏览器直观的查看实例状态进行实例操作^[1]。

为了实现自动化部署, 按需分配资源, 方便扩充和变更, 本系统选择在云计算的 IaaS 基础上实现 CDN 和 Hadoop 的基本功能, 实现云计算的承载软件 OpenStack 与 CDN 的通信, 让云计算能够监控 CDN 的负载, 实现资源的按需分配, 智能调度, 自动部署。

2.2 CDN 系统

CDN 即内容分发网络, 主要为访问者提供网络应用的就近访问, 通过中心服务器来分析不同区域部署的边缘服务器与用户的关系, 通过一定的负载均衡策略和缓存技术, 将用户请求重定向到距离用户最近并能快速提供服务的边缘服务器上, 架构如图 2 所示。结合云计算按需服务弹性调配的功能解决传统 CDN 基础设施资源利用率低的问题, 使其能更好的缓解网络拥堵, 提高用户访问速度^[6]。

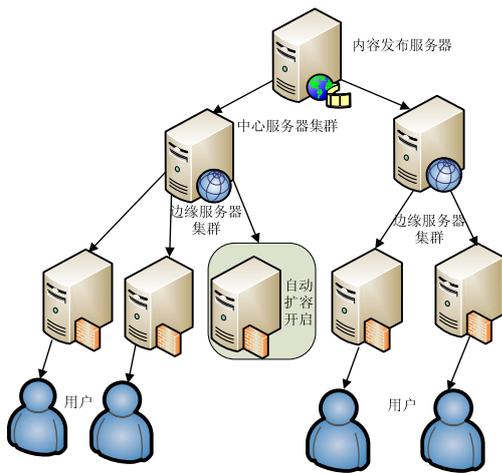


图 2 CDN 架构图

2.2.1 负载均衡策略

CDN 中负载均衡技术主要分为本地负载均衡和全局负载均衡两类。

本地负载均衡主要是将用户请求转发到可以快速提供服务的缓存服务器。本系统主要针对 HTTP 的音视频流媒体业务的分发加速，采用渐进式下载方式，边播放边下载，TCP 连接时间较长，一个用户请求可能长时间占据某一台缓存服务器^[7,8]。因此采用 Nginx 作为本地负载均衡服务器，作为一个区域内所有用户请求的入口，采用 Least Connected 调度算法，把用户请求均衡的转发到节点内部缓存服务器集群连接数较少的 Squid 缓存服务器。

全局负载均衡主要根据测速模块得到的 IP 对应的最快访问的节点，将用户请求 HTTP 重定向到最近的 CDN 节点。HTTP 重定向技术通过在授权的 DNS 里直接将域名解析为 GSLB 的 IP 地址，用户访问 GSLB，GSLB 根据用户的真实 IP 地址来判定地理位置。

2.2.2 缓存策略

CDN 系统中的边缘节点主要作为资源分发的代理服务器，用来缓存常用资源，就近为用户提供服务。加快资源响应速度节省传输消耗和后台服务器压力。但是，边缘服务器本身磁盘空间有限，想要高效的缓存数据就需要在流媒体对象耗尽空间前及时清除旧数据。

传统的缓存替换算法主要根据对象的缓存效用值，替换时将效用值最小的媒体对象替换掉，优秀的缓存替换算法在考虑目标价值的同时还要体现用户的访问

特点，选择高效的替换算法能够明显提升媒体分发效率。传统的缓存替换算法有：LRU(Least Recently Used)、LFU(Least Frequently Used)等。

然而，媒体对象大都是有严格顺序的大文件，单靠传统算法无法很好的将用户的喜好因素利用起来处理流媒体服务。针对此类文件，可以利用用户对于流媒体文件的访问倾向性采用分段缓存的方法。这种倾向性主要分为两类：文件间的倾向性，主要表现在媒体文件被访问的频率不同；文件内部的倾向性，主要表现在媒体文件在不同部分被用户访问频率不同。传统的研究工作主要对文件间倾向性进行相关研究，其中 Zipf 分布被广泛地用来描述这种特性：

$$\sum_{i=1}^N P_i = 1, \lambda = \frac{1}{\sum_{i=1}^N \frac{1}{i^{1-\alpha}}}, P_k = \frac{\lambda}{k^{1-\alpha}}$$

如上公式中 N 为媒体资源的总数，i 为 N 个资源按流行度从高至低排序后资源的序数。从 Zipf 分布的规律可以得出二八定律，即 80% 的用户只访问 20% 的资源内容。由此可将 20% 的资源定义为高热度的资源，80% 资源定义为低热度的资源。本文将针对新媒体互动广播项目中的录制的音频节目以及视频点播资源的热度差异提出不同的缓存策略，如图 3 所示。

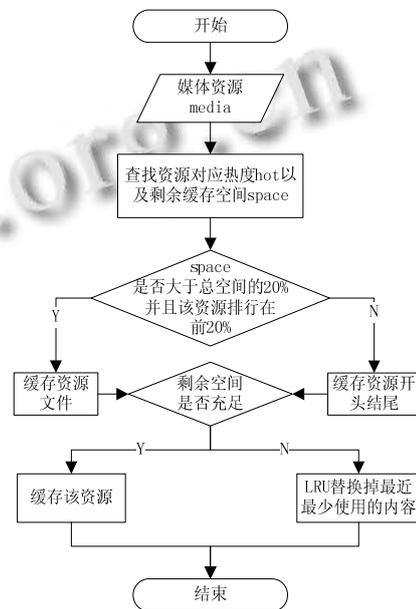


图 3 缓存流程

2.3 客户端测速

现有的 CDN 全局负载均衡技术在为用户分配最优服务器时会先探测各个边缘节点的负载情况和响应

时间,然后再选出合适的边缘节点给用户,这是从服务器端的角度出发,这种探测方式只是代表从中心服务器到边缘节点的速度,而不是从用户到边缘节点的真实速度.因此需要一种从客户端开始测量的主动测速技术^[9].

客户端主动测速技术是为了优化 GSLB 的调度策略而设计的,通过在网页中嵌入测速程序,当用户访问网页时,便会顺带的从各边缘下载一个小文件,然后记录各边缘节点的往返时间,并把 IP 和时间记录到日志中,通过 Hadoop 分析处理,完成后交给 GSLB 服务器,作为同一区域用户下次节点选择的依据.过程如下图所示.

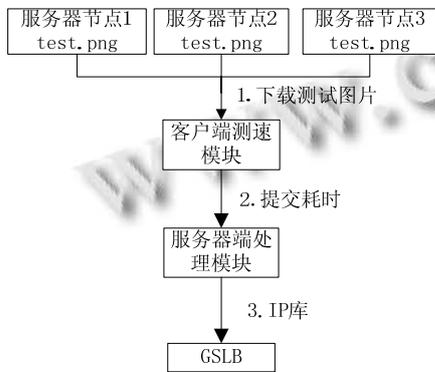


图4 测速过程

① 在网站首页嵌入测速脚本,访问资源首页时,通过 JS 触发测速代码,从配置好的各服务器节点下载一个小文件并记录开始和结束时间,下载完毕后生成“已测速”cookie,提交下载时间给服务器.

② 服务器端生成 IP 库,将 IP 和总时间日志提交给 Hadoop 日志处理系统进行分析.

③ 提取出每个 IP 对应的延时最小的 CDN 边缘节点,并将此节点作为最佳节点,结合归属地和运营商统计出最佳线路,为下一次访问的全局负载均衡提供依据,保证用户访问的是最快的节点.

2.4 Hadoop 分布式处理

为了针对不同区域的用户提供个性化的流媒体服务,完善流媒体分发的负载均衡策略,需要通过对不同区域边缘服务器中的用户访问日志进行分析和挖掘.通过数据清理和转换,得到有用的数据群然后利用数据挖掘技术从中分析出用户的访问模式,在此基础上有针对性的提供个性化的服务.随着项目结构复杂性的不断提高,站点访问量不断增加,需要处理的数据

随之增大,当面临百万级日访问量的海量日志数据时,分布式处理就显得刻不容缓了.

为了处理海量日志数据而设计的 Hadoop 日志处理模块主要利用部署在 CDN 节点中的 Scribe 日志收集器从分布在各地的服务器上搜集用户访问日志,统一交给 Hadoop 服务器集群,利用集群分布式处理海量数据,统计和挖掘有用信息.架构如下图所示.

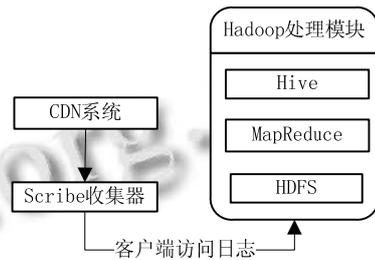


图5 Hadoop 日志处理

2.4.1 MapReduce

MapReduce 框架建立在 HDFS 之上,负责以分布式的形式来处理计算任务.对于体积大的日志文件,单机内存无法处理,利用 MapReduce 通过多台服务器同时处理可以快速分析出所需结果.

① Map

函数 MapReduce 首先根据配置的文件块大小参数,将输入文件切割成 M 份,每一份一般为 16-64MB,以便 Map 任务能以并发方式进行.在日志的处理工作中 Map 过程主要进行数据清洗和用户识别.

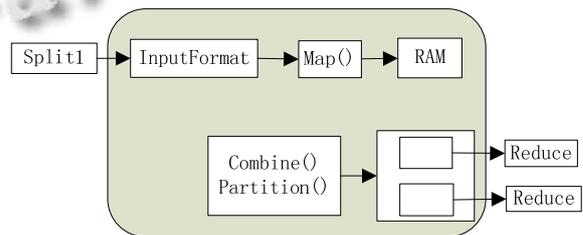


图6 Hadoop 中 Map 操作

Map()方法每次从原始日志文件中读取一行记录,将其按通用日志数据对应的格式和对应域进行划分,提出各个域中的数据,分析此记录中的各项信息,根据定义的数据清洗原则过滤掉无效的记录.

针对日志中已经识别的用户,使用已经分配好的 clientID 作为 userid 进行处理,并作为 key,将用户访问

时间和访问 url 共同组成一个(url,time)作为当前用户 userid 对应的 value.

② Reduce

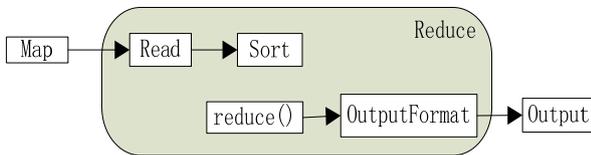


图 7 Hadoop 中 Reduce 操作

将 userid 对应的(url, time)搜集排序后, 整理为 <key=userid,listof(value=(url,time))> 后, 传递给 Reducer 进行合并处理后输出, 将用户访问的 IP 与主动测速结果, 资源和访问情况利用 Hive 数据仓库中的数据表记录下来进行管理. 以备查询能够最快为用户提供访问服务的 url 所对应的边缘服务器 IP. 对于流媒体热度的统计过程也与此类似.

3 实验与结果分析

目前新媒体广播项目中主要涉及的是音频流的分发, 该实验中主要对音频资源的分发做测试. 通过虚拟化技术, 使用 OpenStack 将 4 台计算机, 共 16G 内存, 600G 硬盘, 虚拟出 12 个计算单元搭建云操作系统, 用于搭建和管理 CDN 系统进行仿真实验. CDN 系统中选用 Ubuntu 14.04 作为操作系统、Squid2.7 作为缓存服务器、Nginx1.5.12 作为负载均衡和反向代理服务器.

通过测试得到的在相同服务器配置、响应时间不超过 1000ms 的情况下, 该云 CDN 系统与传统 CDN 的边缘节点资源访问最大并发量对比结果如下表所示. 从中可以看出相较于传统 CDN 系统, 本文所述的云 CDN 系统能够承受更多的并发访问.

表 1 传统 CDN 与云 CDN 最大并发量对比

文件类型	传统 CDN		云 CDN	
	同时上线并发量	步进并发量	同时上线并发量	步进并发量
音频	150	700	300	900

另外, 分别利用测试元素对三层结构中的源站、中心节点、边缘节点进行响应时间和并发测试. 首先根据热度排行在边缘节点上预存 N 个资源文件, 然后分别模拟单个用户访问; N 个用户同时访问一个文件, 稳定运行一定时间; N 个用户分别步进访问 N 个文件; 三种情况进行测试^[10-11]. 测试结果如下.

表 2 音频并发量与响应时间统计

	1:1	N:1		N:N	
		并发量	响应时间 (ms)	并发量	响应时间 (ms)
边缘节点	9.6ms	300	536.45	900	807.32
中心服务器节点	15.31ms	150	563.06	700	806.36
发布服务器节点	19.30ms	100	582.21	400	563.25

以上测试结果中, 本 CDN 系统不同层次中服务器节点的资源响应时间和并发量的对比可以看出, 通过本 CDN 系统的分发处理, 将用户请求重定向到合适的边缘节点提供资源服务时能够达到相对最理想的响应时间和并发量.

综合以上实验结果可知本文中设计的云 CDN 系统能够达到提高系统的资源访问并发量、提高资源响应速度, 改善系统性能的效果.

4 结语

本文通过对新媒体广播项目中流媒体分发需求进行分析, 结合云计算与 CDN 相关技术, 设计了基于云计算的流媒体分发系统, 利用客户端测试技术和媒体热度分析来提高传统的负载均衡和缓存效率, 进而提升系统的流媒体分发能力. 通过实验证明了该系统相比于传统的流媒体分发系统在响应速度、和并发处理方面的能力有了明显的提高. 后续仍需要不断的完善分发系统的功能, 包括视频资源的处理和自适应分发功能.

参考文献

- 1 谌丽. 基于云平台的 CDN 平台研究及其缓存技术探索[硕士学位论文]. 广州: 华南理工大学, 2012.
- 2 孙连生, 王雷, 黄承真. 面向流媒体的 Hadoop 调度策略研究. 小型微型计算机系统, 2014, 35(4): 728-733.
- 3 尹浩, 詹同宇, 林闯. 多媒体网络: 从内容分发网络到未来互联网. 计算机学报, 2012, 35(6): 1120-1130.
- 4 雷起. 基于云平台的多约束流媒体内容分发方法研究[硕士学位论文]. 郑州: 郑州大学, 2014.
- 5 肖颖, 卢春红. 基于云计算的多路网络流媒体分布式最优存储与分配策略. 计算机应用研究, 2015, 32(6): 1835-1838.
- 6 常龙. 基于云计算的移动流媒体平台的研究与实现[硕士学位论文]. 沈阳: 东北大学, 2014.

- 7 Wang F, Liu JC, Chen MH. CALMS: Cloud-assisted live media streaming for globalized demands with time/region diversities. Proc. IEEE INFOCOM, 2012, 131(5): 199-207.
- 8 Gregoire JC, Hamel AM. On scheduling live media streaming in the cloud—A study. IEEE International Symposium on a World of Wireless, Mobile & Multimedia Networks. Sydney. IEEE. 2014. 1-6.
- 9 蒋业文.CDN 关键技术的研究及其系统设计[硕士学位论文].广州:华南理工大学,2014.
- 10 吴金福,田野.中国大陆 CDN 的测量方法研究与性能分析.计算机工程与科学,2014,36(4):620-626.
- 11 张兴军,钱德沛,伍卫国,张文杰,何戈.内容分发网络性能测量方法研究与实现.小型微型计算机系统,2005,26(1):1-5.

www.c-s-a.org.cn

www.c-s-a.org.cn