

集群环境下 CMS 的核心架构设计

祁晖, 王佳

(长春理工大学 计算机科学技术学院, 吉林 长春 130000)

摘要: 通过对非集群环境与集群环境的比较分析, 提出了一种适合集群环境的 CMS 最核心部分(发布、查看新闻, 上传、下载新闻附件)的架构设计, 它是对非集群环境架构设计的改良。通过将文件存储在数据库服务器中, 而应用服务器存储文件的副本, 从而解决了非集群环境的架构移植到集群环境时导致文件无法正常更新的问题。最后通过引入缓存机制, 使得这种架构设计在性能方面有了较大的改进。

关键词: 集群; 内容管理系统; 新闻发布系统; 架构; 缓存

中图分类号: TP39

文献标识码: A

Core architecture design of CMS in cluster environment

QI Hui, WANG Jia

(School of Computer Science and Technology, Changchun University of Science and Technology, Changchun 130000, China)

Abstract: By comparing non-cluster environment with cluster environment, an architecture design of the core part of CMS (publish and view news, upload and download news accessory) is put forward in this paper, which is suitable for cluster environment and is the improvement of the architecture design for non-cluster environment. The problem that files can not be normally updated, which occurs when the architecture for non-cluster environment is moved to cluster environment, is solved by restoring files into database servers and restoring copies of files into application servers. Finally, the performance of the architecture design is greatly improved because of caching mechanism.

Key words: cluster; CMS; news publishing system; architecture; cache

CMS 的应用相当广泛, 大到门户或商业网站, 小到个人网站。开源的 CMS 也非常多, 如 OpenCMS、Infogluce 等。一个功能强大的 CMS 可以对内容进行分类管理, 定制内容属性, 定制界面, 增加、删除、修改、查看内容, 增加、删除、查看附件等。笔者曾经开发过一个具备以上功能的 CMS, 此系统所使用的技术与 Infogluce 十分相似, 都是基于 J2EE 技术构建的。这个 CMS 先后被 2 个政府网站所使用, 运行在单应用服务器环境, 到目前为止一直运行稳定。系统的核心部分, 即内容管理和附件管理部分还被集成到一个职称评审系统中, 起初在测试环境(单应用服务器环境)下运行还很稳定, 但被部署到生产环境(集群环境)时就出现了很多问题, 因此需要对系统的架构进行重新设计。

1 非集群环境下 CMS 的核心架构

非集群环境系统结构如图 1 所示。这是 1 个非常典型的单服务器环境, 由 1 台 Web 服务器、1 台应用服务器和 1 台数据库服务器组成。CMS 主要被部署到应用服务器上。

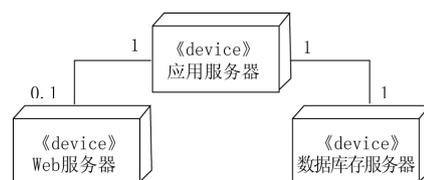


图 1 非集群环境的系统结构图

一条新闻通常由标题、作者、更新日期、正文以及附件等部分组成。一种设计是将所有这些信息都保存到数据库中, 当用户查看新闻时, 需要从数据库中读取新闻的所有信息。这种设计有 2 个非常大的缺点: (1)对

于新闻正文、附件等大数据量的信息(正文长度通常在几KB到几十KB之间,附件更大),在传输到用户浏览器的过程中必须从数据库服务器传输到应用服务器,然后再传输到Web服务器,最后传输到浏览器。其传输路径长,经过的节点多,当用户访问量大的时候,响应用户的时间长。(2)不利于建立全文检索。由于正文或附件等是以字节流的形式存储在数据库服务器中,而数据库并不提供针对字节流数据的搜索功能,因此,几乎不可能为正文或附件建立全文检索。

一种改进的设计是将新闻正文和附件以文件形式存储在应用服务器的磁盘中,而新闻的其他信息则存储在数据库里。这样,用户在查看新闻或者下载附件时,获取正文、附件这些信息就不需要进行数据库操作,而直接从应用服务器磁盘中读取文件,从而减少了应用服务器与数据库服务器之间的通信量,提高了系统的响应速度。此外,还可以对正文和附件文件建立全文检索。这种设计也有缺点,由于存储在应用服务器的可靠性不如存储在数据库服务器高,因此,需要经常备份应用服务器中存储的正文、附件等文件。相比于对系统性能的提高和提供全文检索功能来说,代价还是比较小的,因此笔者开发的CMS采用第二种设计。

2 系统部署在集群环境下时遇到的问题

职称评审系统在正式使用时需要部署在集群环境中,集群环境的系统结构如图2所示。



图2 集群环境的系统结构图

集群环境通常有多台Web服务器、多台应用服务器和有多台数据库服务器。集群环境能够提供更高的性能、更可靠的服务。CMS主要部署在应用服务器集群上,每台应用服务器上都有一个独立的CMS。假设如图3所示情形,设A、B两台应用服务器的时间是同步的。

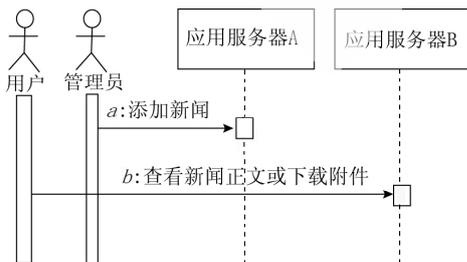


图3 系统时序图

a时刻: 管理员请求添加新闻, 应用服务器集群中

只会有一台服务器提供服务, 即应用服务器A。此时, 应用服务器A中就保存了新闻的正文文件、附件文件, 而数据库服务器中保存了新闻的标题、作者和更新时间等。

b时刻: 用户请求查看新添加的新闻, 请求被路由到应用服务器B, 此时, 可以从数据库中读取新闻的标题、作者和更新时间等信息, 但是新闻的正文和附件却无法获取, 因为正文和附件存储在应用服务器A上, 而集群环境下各台应用服务器是相对独立的, A服务器的改变并不会影响到B服务器。为此, 需重新设计系统的结构, 以适应集群环境。

3 系统的重新设计

系统在集群环境下遇到的问题主要是由于正文和附件只保存在应用服务器上, 而各个应用服务器之间又是互相独立的, 不能得到同步更新。为解决这个问题, 决定将正文和附件保存到应用服务器上的同时, 保存到数据库服务器上, 在数据库中还必须存储应用服务器上正文和附件的文件更新时间。当用户请求查看新闻正文或下载附件时, 系统先在应用服务器上查找该文件, 如果文件不存在, 则到数据库服务器上查找, 并将文件从数据库服务器导入到应用服务器上, 然后再将应用服务器上的文件传输到浏览器; 如果文件存在应用服务器上, 此时是否可以直接将应用服务器上的文件传输到浏览器? 这是一个需要进一步讨论的问题。假设存在如图4所示一种情况, 设A、B两台应用服务器的时间是同步的。

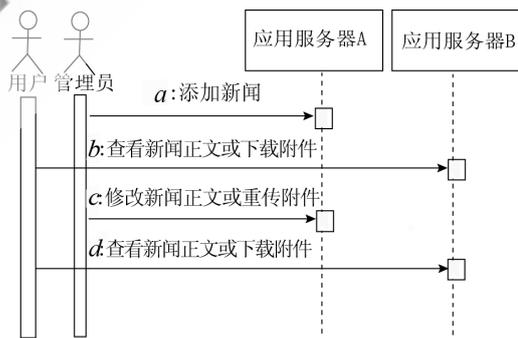


图4 系统时序图

a时刻: 管理员请求添加正文, 此时正文文件被保存到应用服务器A上, 同时保存到数据库服务器上, 数据库中记录的正文文件更新时间是a。

b时刻: 用户请求查看管理员在a时刻添加的正文, b必定大于a。用户被路由到应用服务器B上, 由于B上并不存在正文文件, 于是系统到数据库服务器上查找正文, 并将正文从数据库服务器传输到应用服务器B上, 保存成正文文件。此时, 应用服务器B上就记录了正文文件的更新时间b。

c 时刻：管理员请求修改正文，假设 $c > b$ 。管理员被路由到应用服务器 A 上，此时应用服务器 A 上修改了正文文件，同时数据库服务器上修改正文文件的更新时间为 c 。

d 时刻：用户请求查看正文，假设 $d > c$ 。用户被路由到应用服务器 B 上，此时 B 上已经存在正文文件。如果直接将这个正文文件传输给用户，则用户看到的是管理员在 a 时刻添加的正文，而管理员在 c 时刻修改了正文，因此用户看到的并不是最新的正文。

解决方案如下：如果系统能够在应用服务器上查找到正文文件，则需要将正文文件的更新时间与数据库中记录的正文更新时间进行比较，如果前者大于等于后者，则证明应用服务器中的正文文件是最新的，可以直接传输给用户；否则，证明在应用服务器载入正文文件之后管理员更新了正文，应用服务器中的正文文件不是最新的，如上例中 $b < c$ (b 是应用服务器 B 中正文文件的更新时间， c 是数据库中记录的正文的更新时间)。此时，需要重新从数据库服务器载入正文文件。用户请求查看新闻正文或下载附件时，系统的活动如图 5 所示。

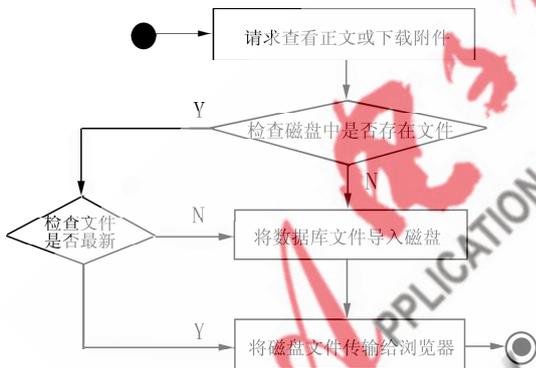


图5 系统活动图

4 系统性能的提高

系统经过重新设计之后，已经可以在集群环境下正常地运行。但是，当用户请求查看新闻正文或请求下载附件时，假定系统能够在应用服务器上直接命中正文文件或附件文件(绝大多数情况下也正是如此)，系统还需要比较文件在应用服务器上的更新时间和数据库服务器中记录的文件更新时间，这样就需要一次数据库查询操作。用户每查看一次新闻正文或请求下载附件就需要查询一次数据库，增加了数据库的负担。为了减少数据库访问次数，可以为系统添加缓存组件，用于缓存数据库中的数据。对于 CMS 来说，缓存中存储的就是新闻对象和附件对象，前者包含标题、作者和正文更新时间等属性，后者包含文件名和文件更新时间等属性。缓存组件被添加到系统后，当用户请求查看新闻正

文或下载附件时，缓存组件的活动如图 6 所示。

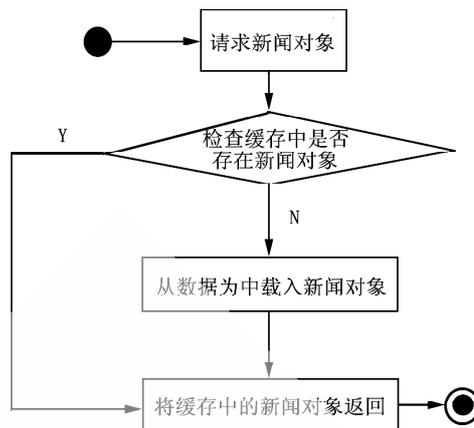


图6 添加缓存组件后的系统活动图

当需要比较文件在应用服务器上的更新时间与数据库中记录的文件更新时间时，就向缓存组件请求新闻对象或附件对象。图 6 所示为请求新闻对象时缓存组件的活动，请求附件对象时缓存组件的活动与此类似。

使用缓存容易产生脏数据。为了保证缓存中的对象能够得到更新，可以设定缓存的过期时间，例如 5 min。经过这番改进，能够有效地减少数据库的访问次数，减轻数据库的负担，提高系统的响应速度。

本文所介绍的 CMS 核心架构设计可以用于构建大型的门户或商业网站，能够在集群环境下稳定运行。当用户访问量增加时，并不会明显增加数据库服务器的负担，系统的负载能力主要和应用服务器集群中的服务器数量有关。由于系统的所有数据都存储在数据库服务器中，应用服务器中的文件只不过是数据库中数据的副本，因此数据的可靠性和安全性都有了很大的提高，同时，还可以对应用服务器中的文件建立全文检索以提供更强大的搜索功能。系统稳定运行的一个前提是应用服务器集群中的各个服务器的时间必须同步，目前已经有技术可以保证这一需求。

参考文献

- [1] ARLOW J, NEUSTADT I. UML 2 and the unified process[M]. Addison-Wesley Professional, 2005.
- [2] 孙卫琴.精通Hibernate—Java对象持久化技术详解[M].北京:电子工业出版社,2005.
- [3] 吴少刚,陈晓玲.J2EE应用服务器集群性能研究[J].计算机工程与设计,2007,28(18):4410-4412.
- [4] 丛林,杨扬,李晓东,等.基于企业服务总线的内容管理系统的研究应用[J].计算机应用研究,2007(1):255-257.
- [5] 夏纯中.轻量级企业内容管理系统的设计与实现[J].计算机工程与设计,2007,28(17):4233-4236.

(收稿日期:209-05-11)