Postgresql 数据库集群在主题网络爬虫的应用①

刘淑梅　夏　亮　许南山　(北京化工大学 信息研究院 北京 100029)

摘　要： 根据网络信息量大的特点，以及主题网络爬虫效率上的要求，将 postgresql 数据库集群技术运用在主题网络爬虫当中，解决了爬虫对大信息量的存储，并采用缓存技术解决了集群技术在爬虫应用中的效率瓶颈。

关键词： 网络爬虫；搜索引擎；主题相关；遗传；抓取

Topic Spider with Postgresql Datebase Cluster

LIU Shu-Mei, XIA Liang, XU Nan-Shan

(Academy of Information, University of Chemical Technology, Beijing 100029, China)

Abstract: In respect to the characteristics of hugeness of net information and request for spider efficiency in topic net, this paper applies postgresql database cluster to the topic net spider, meets the need for huge storage space by spider, and also tackled the bottleneck of efficiency with cache technology when the cluster technology is applied in spider..

Keywords: spider; search engine; database; postgresql; cluster

1　引言

搜索引擎的重要性随着互联网的快速发展越来越重要，在庞大的网络信息中寻找需要的信息，搜索引擎是最有效的工具。同时对于搜索引擎的主题化，个性化要求也越来越高。作为主题搜索引擎后台数据抓取平台—网络爬虫也成为一个急需解决的难点。主题爬虫的目的就是抓取更多的主体网页，但是抓取得网页越多，对硬件资源的需求也就越高，按照以往的单线做法已经不能满足抓取到的信息量的存储[1-3]。本文提出一个基于 postgresql[4-6]数据库集群的方式，作为主题爬虫的后台储备，满足主题爬虫大量信息的抓取。

2　主题爬虫框架

一个的主题爬虫的框架由四部分组成，控制器，解析器，主题分析器，存储器。结构图如图 1。

控制器的功能是给爬虫的各个线程分配任务，我们的爬虫采用的是多线程技术，各个线程间的任务分配需要由控制器来统一安排。

解析器的功能是抓取页面并解析页面信息，提取页面的链接地址与对应的链接文本。

主题分析器是对链接文本进行计算，判断链接地址的相关性，将主体相关的连接地址返回给控制器，由控制器统一管理。

链接地址库存放相关性权重超过阀值的连接地址。

数据抽取的功能是从爬虫抓取来的网页信息中按类抽取出有用信息，并剔除噪音信息。

资源库用来存储按类抽取并被剔除了噪音的信息。

① 收稿时间:2010-03-31;收到修改稿时间:2010-05-11

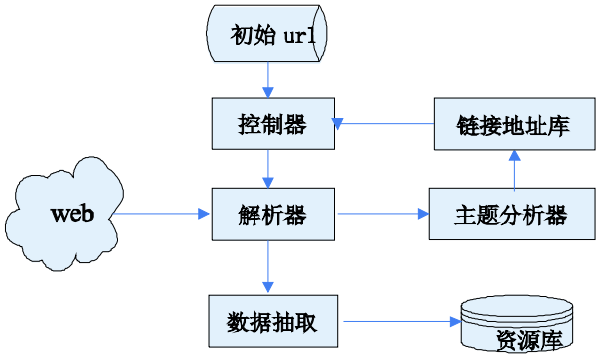

图 1 爬虫框架图

3 postgresql数据库集群在主体爬虫中的应用

根据中国互联网络信息中心统计的信息，到2008年底，中国的网页数量超过了160亿个，按照每个网页为50k来计算，那就是大约为76万G的信息量，全世界的网页信息量就是比这个还要非常大的多。那么作为互联网数据采集的工具主题网络爬虫就非常的需要一个强大的后台数据存储系统来支持它。

3.1 postgresql 数据库集群的实现

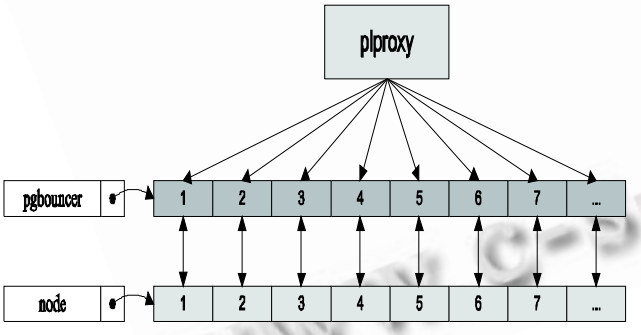

图 2 主题网页链接图

Postgresql 集群[7]可以借助 plproxy[7]来实现，在 plproxy 内部可以将后台各个数据库作为一个工作结点，针对 plproxy 的每一个操作都随机性哈希到后台任一个节点上，也就是说 proxy 结点实地上并没有存储数据，数据都在后台数据库中。在 plproxy 结点上维护了一个后台数据库列表，非常灵活的控制数据库集群，不受物理因素的影响。那么接下来传统单一的做法就是采用哈希计算寻找数据库结点，打开数据库，做相关的修改，插入，选择操作，关闭数据库。这对于大数据量来说打开数据库链接和关闭数据库链接操作是非常耗时的。在 plproxy 结点与数据库结点之间采用 pgbouncer[7]链接池技术可以很好的解决这个问题，pgbouncer 在处理一次事务之后将连接放回连接池中，下次需要的时候再从连接池中取一个连接。这样就避免了大量的开启和关闭数据库的操作。结构如图 2 所示。

3.2 postgresql 数据库集群在资源库中的应用

爬虫内部通过一个 url 获取到网页信息之后，解析器先解析其中连接地址以及对应的链接文本，数据抽取器从中按类抽取出剔除了 html 标签的文本数据。这里的类别主要是标题，以及正文，标题往往是概括网页内容最优的部分，正文是内容最全的部分。抽取器存储到资源库的三个必不可少的部分就是 url 连接地址，标题，正文。对于抽取器而言它只和 plproxy 结点的主机通信，它不用考虑 plproxy 之后的集群。下面是 plproxy 接到消息之后的流程。

① 对 plproxy 发出插入请求。

② Plproxy 寻找通过 plproxy language 自定义的函数。

③ 通过自定义函数以及 plproxy 的三个配置函数找到集群配置信息。

④ 根据自定义函数中 RUN ON hashtext(url) %n 来随机分配到后台 n 台服务器中。

⑤ 从后台哈希到的服务器连接池中取一个数据库链接，对数据库做插入操作。

⑥ 将链接放进连接池。

⑦ 一次插入操作完成。

从上面可以看到 Plproxy 接到一个消息，不是对其后面的每一台机子都进行同样的插入操作，它先对 url 做哈希再与 n 做与操作寻找一个目标机器。每一个 url 在整个过程中走的是直线型路线，但是对全部的 url 来说又走的是不同的路线，分叉点在 plproxy 结点。之后走的路线与服务器的数量 n 和 url 相关。

如果有 M 个有效 url 连接地址，那后台每台服务器上的信息量大概就是(M/n)*25k(剔除标签后的网页平均大小大概是 25k)。也就是说以前一台服务器需要承受 M*25k 的信息量，现在被 n 台服务器分别承受，大大减轻了服务器的数据压力，有效地支撑了爬虫的稳定性。

如图 3 所示，将数据库集群每一个数据库依序排列在一列桶中，桶在列表中的序号为唯一标识，proxy 借用 hash 函数为数据抽取器传回的数据选取目标桶。采用 hash 取桶的好处是在多个进程同时运行的情况下可以将大量的数据平均分配到各个桶中。

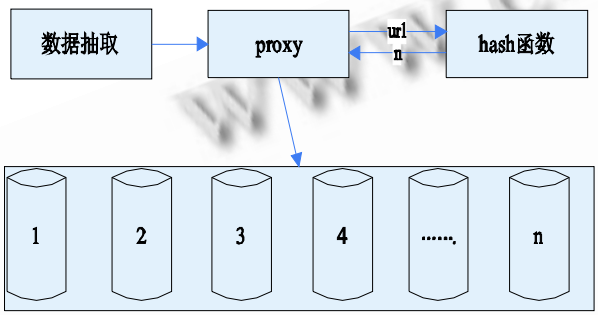

图 3　资源库存储过程图

3.3 postgresql 数据库集群在连接地址库中的应用

在爬虫系统中，连接地址库址也需要数据库集群技术。对于爬虫系统来说，资源库与连接地址库数据库功能不尽相同，在链接地址库中，主题分析器需要往里面不断的插入有效的链接地址，同时控制器还要不断地从链接地址库中取出连接地址用来抓取。也就是说在这里不但有插入数据的动作，还有获取数据的动作。

主题分析器通过对链接文本的计算来获取链接地址的相关性权重，当权重超过阀值的时候往链接地址库中插入。在链接地址库保存有链接地址与其对应的相关性权重。因为不同的网页的 url 不同，所以根据 url 做哈希计算的键值。由 url 来判断存储到哪台服务器。这个过程与资源库的过程相同。

控制器获取待抓取 url 的过程也是通过 plproxy 来分发完成，每次从链接地址库中选择权重最大的 url，因为权重大，说明相关性高，优先度也就高。从链接地址库中选择权重高的 url 就不是随机去选择一台服务器，因为 url 插入到链接地址库中是随机的，也就不知道哪台服务器中的 url 权重是最大。最有效的算法就是每台服务器都选择权重最大的一个 url，接着在 plproxy 中将这些各个服务器中 url 权重最大的排序，再取权重最大的一个 url，这样就能够保证控制器每次取得是权重最大值。这里有一个效率的瓶颈，假设有 m 台服务器，每次选择都要做 m+1 次排序，只选择最大的一个 url，速度将会大大地降低，对于爬虫来说是得不偿失。采用缓存的方式可以很好的解决这个瓶颈，这个方式就是每次选择多个 url 缓存起来提供给控制器使用，等缓存中的 url 用完之后再从链接地址库中选择，如图 4 所示。现在假设每次选择 n 个 url，后台每台服务器每次选择 n 个 url。将选择到的 n*m 个 url 放在 plproxy 中进行排序，选择权重最大的 n 个，缓存起来。没缓存之前需要 n*m+1 次排序，缓存之后只需要 m+1 次。大大降低了排序次数，提高了效率。流程如下。

① 控制器对 plproxy 发出选择请求。

② Plproxy 寻找通过 plproxy language 自定义的函数。

③ 通过自定义函数以及 plproxy 的三个配置函数找到集群配置信息。

④ 根据自定义函数中 RUN?ON?ALL 来分配到后台每台服务器中。

⑤ 从后台每台服务器连接池中取一个数据库链接，对数据库做选择操作。

⑥ 将链接放回连接池。

⑦ 在 plproxy 结点处对从各个服务器中选择的结果进行排序整合，取权重最大的 n 个返回给控制器。

⑧ 一次选择操作完成。

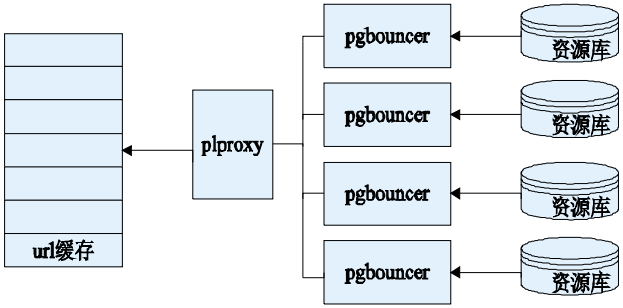

图 4 连接地址库存储图

4 结束语

本文在主题网络爬虫使用的 postgresql 数据库集群技术很好的支持了网络爬虫的存储需求，同时对待抓取的 url 管理上也起到很好的管理作用。

在连接地址库中采用的缓存技术是减少选择排序次数来获取时间的，这里的不足之处在于控制器每次要等到缓存中没有 url 时，才再从后台服务器中选择，在这期间链接地址库还不断的从主题分析器中得到 url 及其权重，这时就有可能出现新入库的 url 权重要比缓存中的高，而造成权重高的 url 没有被及时的抓到。下一步的研究重点就是怎样提高速度的同时满足权重高的 url 能及时地被抓取。

参考文献

1 李勇,韩亮. 主题搜索引擎中网络爬虫的搜索策略研究.计算机工程与科学, 2008.30(3):4－6.

2 宋宇,孟祥增. 主题蜘蛛的设计与实现.郑州大学学报, 2007,39(2):42－45.

3 刘金红,陆余良. 主题网络爬虫研究综述.计算机应用研究, 2007,24(10):26－29.

4 陈琚. PostgreSQL 在时空数据管理中的应用.测绘通报, 2008,7:44－46.

5 张爱国,邬群勇,王钦敏. 基于 PostgreSQL 数据库的 GML 数据存储.测绘科学, 2008,3(1):194－196.

6 战疆,冯月利,王珊. PostgreSQL 中文全文索引技术研究与实现.华中科技大学学报, 2005,33:213－216.

7 配置一个使用 plproxy 的 PostgreSQL 数据库集群, [2008-5-25]http://www.pgsqldb.org/mwiki/index.php