走向架构师之路

个人资料

cutesource

访问： 2029600次
积分： 17483
等级： BLOG 7
排名： 第287名

原创： 152篇 转载： 14篇
译文： 0篇 评论： 901条

文章搜索

文章分类

土鳖混外企 (2)
源码分析 (16)
工作心得 (40)
技术积累 (63)
构建高性能web系列 (4)
架构分析 (6)
移动云 (41)

文章存档

2014年10月 (1)
2013年10月 (1)
2013年09月 (1)
2013年08月 (1)
2013年05月 (5)

展开

阅读排行

单点登录SSO的实现原理 (129849)
JVM学习笔记（一）------ (84431)
JVM学习笔记（二）------ (65330)
探索WebKit内核（一）-- (57228)
分布式设计与开发（一）

分布式设计与开发（二）------几种必须了解的分布式算法

标签： 算法 文档 server blog 集群 优化

2010-08-15 11:53 47561人阅读 评论(4) 收藏 举报

分类： 技术积累（

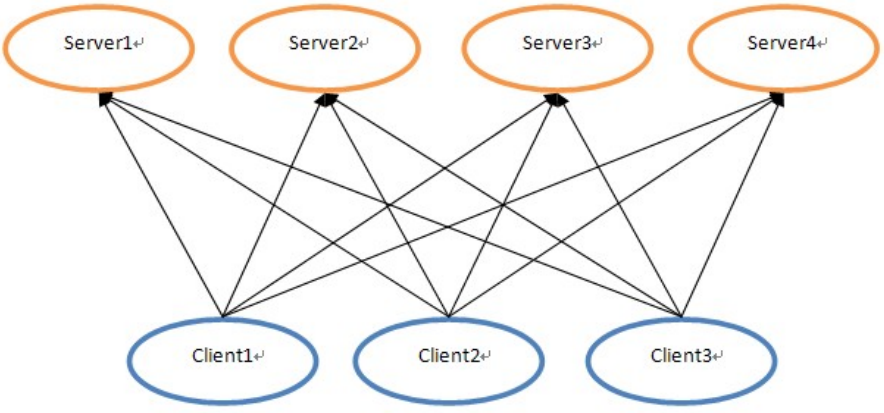
分布式设计与开发中有些疑难问题必须借助一些算法才能解决，比如分布式环境一致性问题，感觉以下分布式算法是必须了解的（随着学习深入有待添加）：

- Paxos算法
- 一致性Hash算法

Paxos算法

1）问题描述

分布式中有这么一个疑难问题，客户端向一个分布式集群的服务端发出一系列更新数据的消息，由于分布式集群中的各个服务端节点是互为同步数据的，所以运行完客户端这系列消息指令后各服务端节点的数据应该是一致的，但由于网络或其他原因，各个服务端节点接收到消息的序列可能不一致，最后导致各节点的数据不一致。举一个实例来说明这个问题，下面是客户端与服务端的结构图：



当client1、client2、client3分别发出消息指令A、B、C时，Server1~4由于网络问题，接收到的消息序列就可能各不相同，这样就可能由于消息序列的不同导致Server1~4上的数据不一致。对于这么一个问题，在分布式环境中很难通过像单机里处理同步问题那么简单，而Paxos算法就是一种处理类似于以上数据不一致问题的方案。

2）算法本身

算法本身我就不进行完整的描述和推导，网上有大量的资料做了这个事情，但我学习以后感觉莱斯利·兰伯特（Leslie Lamport，paxos算法的奠基人，此人现在在微软研究院）的Paxos Made Simple 是学习paxos最好的文档，它并没有像大多数算法文档那样搞一堆公式和数学符号在那里吓唬人，而是用人类语言让你搞清楚Paxos要解

(56537)
分布式设计与开发（二）
(47555)
Tomcat源码分析（一）--
(44363)
JVM学习笔记（三）------
(42108)
构建高性能web之路------
(42029)
分布式设计与开发（三）
(40210)

评论排行

迈向架构师的第一步 (77)
分布式设计与开发（一） (42)
单点登录SSO的实现原理 (40)
在AMD的WIN7上搭建IOS (38)
JVM学习笔记（一）------ (28)
JVM学习笔记（四）------ (23)
JVM学习笔记（三）------ (22)
phonegap源码分析（一） (22)
从Jetty、Tomcat和Mina (20)
最后一天 (19)

推荐文章

*Android自定义ViewGroup打造各种风格的SlidingMenu
* Android 6.0 运行时权限处理完全解析
* 数据库性能优化之SQL语句优化
*Animation动画详解(七)——ObjectAnimator基本使用
* Chromium网页URL加载过程分析
* 大数据三种典型云服务模式

最新评论

迈向架构师的第一步
haryhouqin: 加油！
IT土鳖混外企（二）------ 语言障
神的理想: 不错，15000词放口袋
架构师的楷模
youngke: 加油，学习中
JVM学习笔记（二）------Java代
brianway: 思路清晰，几张图画的挺好的，学习了。
探索WebKit内核（一）------ 菜鸟
l360220954: 楼主，可不可以给一个详尽版的编译方法呢，编译了好几次，用了好几种在网上找的方法，有一个错误实在过不去...
探索WebKit内核（一）------ 菜鸟
l360220954: @NiYongYuanDeErDuo:我也要编译这个了，请问你的问题解决得怎么样了呀？
Hessian源码分析（三）------ He
zwllxs: io部分我可以 补上，并且还改写过大量io下的包，扩充了序列化和反序列化模块
从Jetty、Tomcat和Mina中提炼N
yange102688: 假如一个场景，用户上传文件，某些用户网速较慢，同时存在100个这样的用户，如果BIO且最大线程设为1...
单点登录SSO的实现原理
笑破苍穹: 调理清晰
基于libevent, libuv和android Loc
yangchangeu: epoll是同步非阻塞I/O

决什么问题，是如何解决的。这里也借机抨击一下那些学院派的研究者，要想让别人认可你的成果，首先要学会怎样让大多数人乐于阅读你的成果，而这个描述Paxos算法的文档就是我们学习的榜样。

言归正传，透过Paxos算法的各个步骤和约束，其实它就是一个分布式的选举算法，其目的就是要在一堆消息中通过选举，使得消息的接收者或者执行者能达成一致，按照一致的消息顺序来执行。其实，以最简单的想法来看，为了达到大伙执行相同序列的指令，完全可以通过串行来做，比如在分布式环境前加上一个FIFO队列来接收所有指令，然后所有服务节点按照队列里的顺序来执行。这个方法当然可以解决一致性问题，但它不符合分布式特性，如果这个队列down掉或是不堪重负这么办？而Paxos的高明之处就在于允许各个client互不影响地向服务端发指令，大伙按照选举的方式达成一致，这种方式具有分布式特性，容错性更好。

说到这个选举算法本身，可以联想一下现实社会中的选举，一般说来都是得票者最多者获胜，而Paxos算法是序列号更高者获胜，并且当尝试提交指令者被拒绝时（说明它的指令所占有的序列号不是最高），它会重新以一个更好的序列参与再次选举，通过各个提交者不断参与选举的方式，达到选出大伙公认的一个序列的目的。也正是因为有这个不断参与选举的过程，所以Paxos规定了三种角色（proposer，acceptor，和 learner）和两个阶段（accept和learn），三种角色的具体职责和两个阶段的具体过程就见Paxos Made Simple ，另外一个国内的哥们写了个不错的PPT ，还通过动画描述了paxos运行的过程。不过还是那句话不要一开始就陷入算法的细节中，一定要多想想设计这些游戏规则的初衷是什么。

Paxos算法的最大优点在于它的限制比较少，它允许各个角色在各个阶段的失败和重复执行，这也是分布式环境下常有的事情，只要大伙按照规矩办事即可，算法的本身保障了在错误发生时仍然得到一致的结果。

3）算法的实现

Paxos的实现有很多版本，最有名的就是google chubby ，不过看不了源码。开源的实现可见libpaxos 。另外，ZooKeeper 也基于paxos解决数据一致性问题，也可以看看它是如果实现paxos的。

4）适用场景

弄清楚paxos的来龙去脉后，会发现它的适用场景非常多，Tim有篇blog《Paxos在大型系统中常见的应用场景》 专门谈这个问题。我所见到的项目里，naming service是运用Paxos最广的领域，具体应用可参考ZooKeeper

一致性**Hash**算法

1）问题描述

分布式常常用Hash算法来分布数据，当数据节点不变化时是非常好的，但当数据节点有增加或减少时，由于需要调整Hash算法里的模，导致所有数据得重新按照新的模分布到各个节点中去。如果数据量庞大，这样的工作常常是很难完成的。一致性Hash算法是基于Hash算法的优化，通过一些映射规则解决以上问题

2）算法本身

对于一致性Hash算法本身我也不做完整的阐述，有篇blog《一致性hash算法 - consistent hashing》 描述这个算法非常到位，我就不重复造轮子了。

实际上，在其他设计和开发领域我们也可以借鉴一致性Hash的思路，当一个映射或规则导致有难以维护的问题时，可以考虑更一步抽象这些映射或规则，通过规则的变化使得最终数据的不变。一致性hash实际就是把以前点映射改为区段映射，使得数据节点变更后其他数据节点变动尽可能小。这个思路在操作系统对于存储问题上体现很多，比如操作系统为了更优化地利用存储空间，区分了段、页等不同纬度，加了很多映射规则，目的就是要通过灵活的规则避免物理变动的代价

3）算法实现

一致性Hash算法本身比较简单，不过可以根据实际情况有很多改进的版本，其目的无非是两点：

- 节点变动后其他节点受影响尽可能小
- 节点变动后数据重新分配尽可能均衡

实现这个算法就技术本身来说没多少难度和工作量，需要做的是建立起你所设计的映射关系，无需借助什么框架或工具，sourceforge上倒是有个项目libconhash ，可以参考一下

以上两个算法在我看来就算从不涉及算法的开发人员也需要了解的，算法其实就是一个策略，而在分布式环境常常需要我们设计一个策略来解决很多无法通过单纯的技术搞定的难题，学习这些算法可以提供我们一些思路。

上一篇 分布式设计与开发（一）------宏观概述
下一篇 分布式设计与开发（三）------高一致性服务ZooKeeper

我的同类文章

技术积累（62）

- 探索WebKit内核（三）------ ... 2013-05-12 阅读 10721
- 探索WebKit内核（二）------ ... 2013-04-28 阅读 18249
- 从Samples中入门IOS开发... 2013-01-15 阅读 8386
- 从Samples中入门IOS开发... 2013-01-14 阅读 8272
- 从Samples中入门IOS开发... 2013-01-14 阅读 13958
- 从Samples中入门IOS开发... 2013-01-12 阅读 8463
- 从Samples中入门IOS开发... 2013-01-11 阅读 8191
- 从Samples中入门IOS开发... 2013-01-10 阅读 8005
- 从几个sample来学习Java... 2012-12-12 阅读 16470
- 推荐近年来印象最深的几本书 2012-11-08 阅读 7541

更多文章

主题推荐 c语言 分布式 设计 数据 server 集群

猜你在找

数据结构和算法 一个程序员多年的收藏
有趣的算法（数据结构） 一个程序员多年的收藏
以性别预测为例，谈谈数据挖掘中常见的分类算法 一个程序员多年的收藏
数据结构基础系列(3)：栈和队列 有一种收藏叫转载
《C语言/C++学习指南》加密解密篇（安全相关算法） 多年的收藏

查看评论

3楼 AI桐桐 2015-09-18 20:35发表

您好，谢谢这篇帖子，两个算法说的简单易懂，但是看完之后我觉得两个算法和分布式之间有什么关系呢？盼复，谢谢！

Re: 半吊子的程序员 2015-12-25 14:49发表

回复AI桐桐：举个例子：在memcache中使用的就是一致性Hash算法，你部署好了以后，多个memcache节点读取数据，到底是读取那个节点的数据，是算法的体现

2楼 happylucky 2011-11-28 15:18发表

你好，看了你文章之后，我在想，里用云计算机提高复杂网络中社区发现算法的效率，是不是就需要用到分布式算法设计呢。请指点迷津你，本人底子差。

1楼 j_now 2010-11-16 12:31发表

learn more from http://www.cnblogs.com/ychellboy/tag/paxos/

您还没有登录,请[登录]或[注册]

* 以上用户言论只代表其个人观点，不代表CSDN网站的观点或立场

FTC coremail OPhone CouchBase 云计算 iOS6 Rackspace Web App SpringSide Maemo Compuware 大数据 aptech Perl Tornado Ruby Hibernate ThinkPHP HBase Pure Solr Angular Cloud Foundry Redis Scala Django Bootstrap