

基于Linux的健康云平台服务优化^①

李贝贝^{1,2}, 陶耀东¹

¹(中国科学院 沈阳计算技术研究所, 沈阳 110168)

²(中国科学院大学, 北京 110039)

摘要: 移动互联网和健康大数据正在引发医疗健康管理产业发生颠覆性变革, 各级厂商着力构建形成一套基于“互联网+物联网+医疗服务”技术的以 O2O 商业模式为主的云端健康管理平台。“中科慧康”云健康平台主要为健康机构以及用户提供健康管理的平台。本文从平台架构设计和系统优化入手, 从负载均衡, 分布式缓存, 数据库优化, socket 转发服务器等方面设计了良好的云平台以支撑业务发展。实验结果表明, 该系统架构可以满足百万级别的数据转发和用户访问, 应用实践表明了该方案的可行性。

关键词: 移动医疗; Web 系统优化; 负载均衡; 分布式缓存; socket 转发服务器

Health Cloud Platform Service Optimization Based on Linux

LI Bei-Bei^{1,2}, TAO Yao-Dong¹

¹(Shenyang Institute of Computing Technology, Chinese Academy of Sciences, Shenyang 110168, China)

²(University of Chinese Academy of Sciences, Beijing 110039, China)

Abstract: The mobile Internet and health big data is causing big health healthcare management industry disruptive change, the formation of a firm focuses on building levels based on the “Internet + IoT + medical service” technology-based business model to cloud O2O health management system platform. “Zhong Ke Hui Kang” health cloud platform mainly provides health management platform for health institutions and users. This paper discusses in aspects of the platform architecture design and system optimization starting from load balancing, distributed caching, database optimization, socket forwarding server design a good cloud platform to support business development. Experimental results show that the system architecture to meet the one million level of data forwarding and user access, application practice shows the feasibility of the scheme.

Key words: mobile health; Web system optimization; load balancing; distributed cache; socket forwarding server

随着移动互联网^[1]和健康大数据正在引发医疗健康管理产业发生颠覆性变革, 各级厂商着力构建形成一套基于“互联网+物联网+医疗服务”技术的以 O2O 商业模式为主的云端健康管理平台^[2]。

采用个人用户终端采集及信息无线传输技术, 建立在大数据及云平台技术基础上, 实现企业和个人用户全健康管理服务的科学化、智能化、远程化, 通过网络、电话、移动平台等多个终端, 为客户提供持续、规范、科学的检测数据存储、远程医疗咨询、就医分诊服务、健康行为干预等。

业内典型代表: 开云健康, 福满多, 小云健康等。

“中科慧康”云健康平台主要为中医医疗机构, 健康服务机构, 会员营销机构, 体验店营销机构, 健身服务机构, 连锁药店, 美容美体机构以及用户提供健康管理的平台。2016年用户量预计超过10万。本文研究重点是设计良好的云平台服务构架^{[3][7][8]}以支撑业务发展。

1 “中科慧康”总体结构设计

本系统总体分成三部分: 相关 Web 页面获取模块、Web 信息抽取模块、知识表示模块。系统总体框架如图 1 所示。

① 收稿时间:2016-01-31;收到修改稿时间:2016-03-14 [doi: 10.15888/j.cnki.csa.005379]

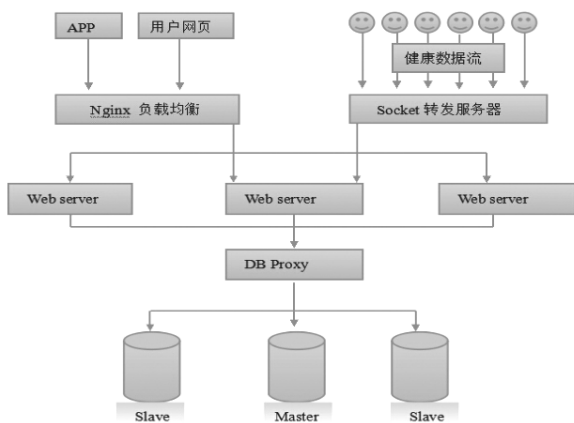


图1 “中科慧康”云平台总体架构图

中科慧康云平台后台利用 spring MVC 和 hibernate 框架以及 JSP 技术, 后台部署于三台服务器, 利用 nginx 做负载均衡到两台服务器上. 用户测量的血压血糖等健康数据通过测量仪器的 GSM 的 socket 通信模块直接连接到 Socket 转发服务器, 转发服务器将数据转发到正式的云平台对数据进行接收处理. 用户和代理商访问网页以及 app 流量入口首先访问前端 nginx 反向代理服务器, nginx 对流量进行分流到后台两台服务器以减轻访问压力.

2 “中科慧康”平台上线遇到的问题

- (1) Socket 转发程序处理缓慢.
- (2) 系统负载很高, cpu 使用率过高, jvm 偶尔出现崩溃情况.
- (3) 网站响应速度较慢.
- (4) 数据库查询缓慢, 数据库经常连接失败.

3 平台服务优化

3.1 socket 转发服务的优化

原来转发程序: 用户上传血压, 血糖数据, 发起 tcp 连接, 服务器监听 9000 端口接收数据, 收到数据以后, 对于数据包格式判定, checksum 判定数据包校验, 数据过滤, 对于正确的数据包进行转发, 转发方式利用 http 连接的 get 方式将数据发送到正式和测试服务器上, 当后台收到数据以后, 返回给转发程序一个 ack 数据包, 表示数据转发成功. 这个过程由图 2 可以描述.

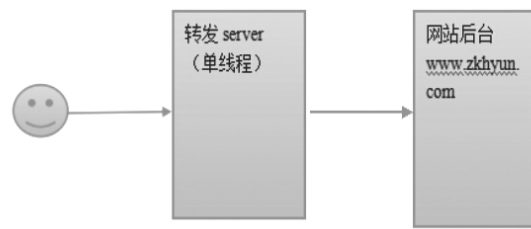


图2 原 socket 服务转发流程示意图

优化后的程序: 利用多线程接收, 引入 java 线程池, 当 9000 端口监听到多个数据同时到来时, 利用多个线程同时处理, 处理完数据以后, 同时转发数据, 性能得到大幅度提升, 云平台根据业务增长量, 用户数量预计到达 10 万, 该程序可以满足 10 万级别用户的转发不会出错. 这个过程由图 3 可以描述.

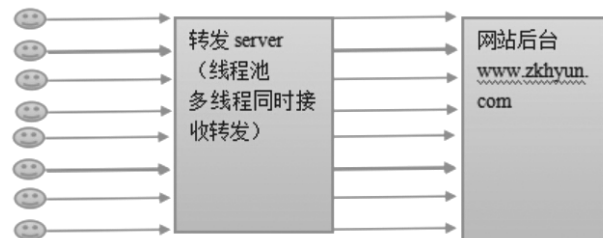


图3 多线程接收, 转发流程示意图

3.2 可以采用 nginx 反向代理服务器做负载均衡^{[6][9]}

对于单个服务器系统负载很高, cpu 使用率过高问题, 因为每天访问量 1 万 pv(page view)造成. Nginx 是一款轻量级的 Web 服务器/反向代理服务器及电子邮件(IMAP/POP3)代理服务器, 其特点是占有内存少, 并发能力强, 中国大陆使用 nginx 网站用户有: 百度、新浪、网易、腾讯等, nginx 默认的分流算法是轮询加权算法.

配置 nginx 服务器核心工作: 在 nginx 的配置文件 nginx.conf 中, 修改配置默认访问 ip 地址为 123.67.154.112: 80 作为流量入口

```

upstream web_app {
    server 115.29.45.167:8080 weight=1 max_fails=2 fail_timeout=30s;
    server 123.67.154.112:8080 weight=1 max_fails=2 fail_timeout=30s;
}

```

配置服务器地址：115.29.45.167:8080
115.29.45.167:8080 同时可以设置访问权重 weight, Fail_timeout 参数可以设置超时访问自动切换到另一个服务器的时间阈值。

配置 Nginx 动静分离, 当请求访问静态页面时, 不必访问网站页面, 直接返回缓存即可, 大大降低网站访问压力

```
location ~.*\.(html|htm|gif|jpg|jpeg|bmp|png|ico|txt|js|css)$
{
    Root /data/www;
    Expires 3d;#缓存时间为 3 天
}
```

转发过程: 当多个 http 请求访问 www.zkhyun.com 时, 按照权重 1:1 将一半流量分流到 123.67.154.112:8080 地址的服务器, 将另一半流量分流到 115.29.45.167:8080 地址的服务器中, 可以使得单个服务器的负载降低一半. 这个过程由图 4 可以描述.

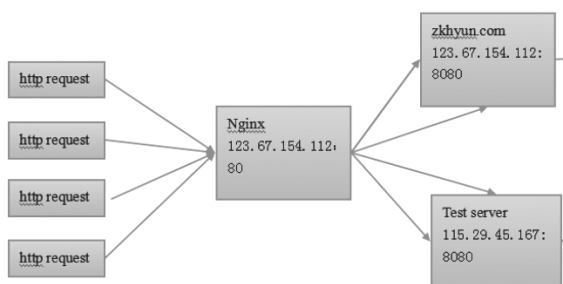


图 4 nginx 反向代理处理流量转发流程示意图

3.3 将 Memcached 应用到健康云平台

由于平台上线, 网站存在访问相应过慢的问题. 根据网站访问的 8-2 原则, 80% 的数据访问会落到 20%, 可以将 20% 的热点数据缓存^{[4][12]}到内存中, 当下次访问到热点数据以后, 不必从数据库中取数据, 而是直接在缓存中得到, 可以大大提高响应速度.

Memcached^[11]是一个高性能的分布式的内存对象缓存系统, 通过在内存里维护一个统一的巨大的 hash 表, 将数据调用到内存中, 然后从内存中读取, 从而大大提高读取速度. Memcached 工作流程示意图如图 5 所示.

Memcached 的工作流程在图 6 中可以看出: 先检查客户端的请求数据是否在 memcached 中, 如有, 直接把请求数据返回, 不再对数据库进行任何操作; 如果请求的数据不在 memcached 中, 就去查数据库, 把

从数据库中获取的数据返回给客户端, 同时把数据缓存一份到 memcached 中; 每次更新数据库的同时更新 memcached 中的数据, 保证一致性; 当分配给 memcached 内存空间用完之后, 会使用 LRU(最近最少使用)策略加上到期失效策略, 失效数据首先被替换, 然后再替换掉最近未使用的数据.

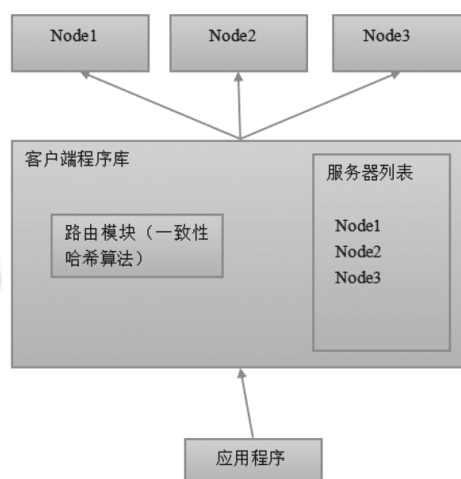


图 5 memcached 工作流程示意图

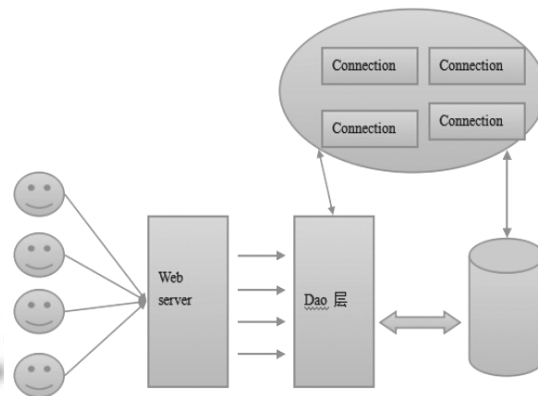


图 6 数据库连接池工作原理

3.4 应用数据库连接池技术

当进行数据库压力测试时, 经常数据 jvm 堆栈溢出. 分析原因发现: 每次建立数据库连接对 DB 进行操作时, 都会产生一个 connection 对象, 当执行完 db 操作以后, 将对象销毁. 当频繁创建和销毁 connection 对象时, 会导致堆的溢出. 加入数据库连接池技术, 当系统运行时, 创建一个连接池, 池中有固定数量的 connection, 如果一个连接需要操作 db, 不是创建一个 connection 对象, 而是直接从池中取出一个 connection 利用, 如果操作完毕, 将 connection 直接还回连接池中, 而不是销毁对象, 这样可以不需要频繁的创建和销毁

connection 对象, 提高数据库的性能. 数据库连接池工作原理如图 6 所示.

4 实验

4.1 Socket 程序测试实验

根据用户测量时间段习惯和每天平台上传数据量分析做出相应的压力测试.

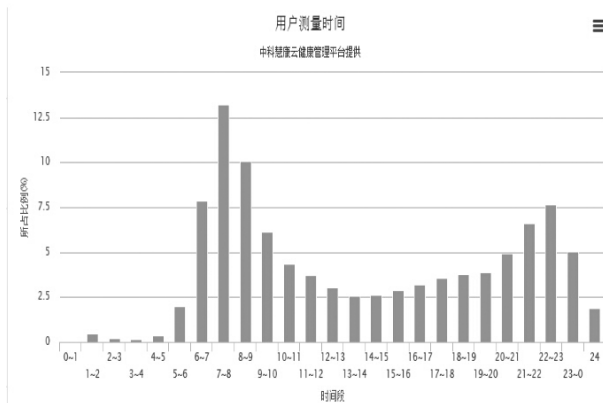


图 7 用户测量时间段统计



图 8 平台每天测量总的的数据量

图 7 描述了用户测试时间的习惯, 显示除了早上 7-8 点会有一个显著波峰, 其他时间段测量数据分布均匀. 图 8 描述了每天平台的接收的总体数据量, 显示总得数据量呈现一个稳定增长的态势. 平台总会员数预计上限在 10w 人, 每人每天测量数据两次, 数据量为 20w 条. 根据测量习惯, 平台的最大并发量为: $20w/24h = 150/min$

对于 socket 进行压力测试的程序, 模拟 20 个线程发送不同的数据量:

```
ExecutorService ex=Executors.newCachedThreadPool();
for (int i = 0; i < 10; i++) {
```

```
ex.execute(new SocketConnect(100));
}
ex.shutdown();
SimpleDateFormat df =new SimpleDateFormat("yydd");
String time = df.format(new Date());
Socket s = new Socket("121.42.32.103", 9000);
DataOutputStream dos = new
DataOutputStream(s.getOutputStream());
dos.writeBytes("5A231180601A00000115111082072" +
time + "115507E7");
dos.flush();
dos.close();
s.close();
return "第 " + id + "个任务还剩" + countDown + "次
连接";
```

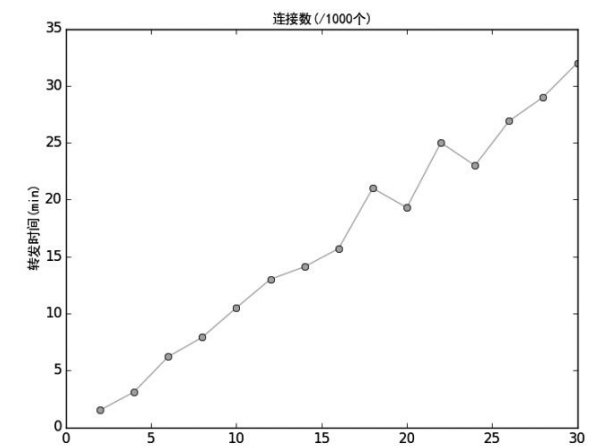


图 9 socket 连接数和转发时间的关系

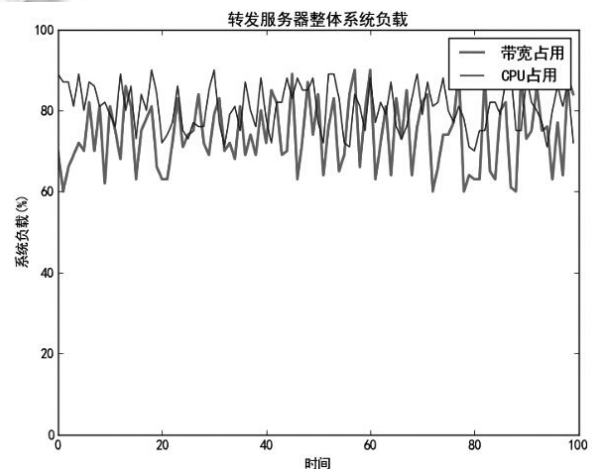


图 10 转发服务器系统负载

图 9 描述了 socket 连接数和转发时间的关系, 从图 9 可以看出, 转发速度在 1000 次/min 的并发量, 要远远大于 150/min 的最大并发量, 所以可以支撑 100w 用户级别的云平台数据量. 图 10 描述了随着连接数的增加, 转发服务器系统负载的变化, 从图 10 可以看出, cpu 使用率和带宽使用率维持在一个稳定的区间.

结论: 该转发服务器可以支撑并发量可以达到 1000/min, 可以满足系统需求, 并且可以满足 120w 数据量的转发.

4.2 网站性能测试实验

利用 http_load 网站压力测试网站对 www.zkhyun.com 进行测试.

```
http_load -p 30 -s 1 urllist.txt
```

命令表示: 向 www.zkhyun.com 并发 30 个线程, 时间为连续 1 秒.

```
sky@sky-Aspire-vs:~/http_load$ http_load -p 30 -s 1 urllist.txt
```

```
20 fetches,30 max parallel,236920 bytes,in 1.0003 seconds
```

```
11846 mean bytes/connection
```

```
19,9941 fetches/sec,236850 bytes/sec
```

```
msesc/connect:22.5552 mean 31.953 max,14.406 min
```

```
msesc/first-response:180.676 mean,390.984 max,17.991 min
```

HTTP response codes:

```
code 200 -- 20
```

测量结果中的主要指标是: 每秒的响应请求连接数 fetches/sec、每次连接平均响应时间 msec/connect 数值, 在云平台优化前和优化后多次测试做对比, 如图 11 所示:

从图 11 可以看出: 优化后比优化前, 每秒相应连接数目增加 4-5 倍, 响应时间显著缩短, 可以更好的满足高并发的访问量, 带来更好的用户体验.

5 结语

本文设计了“中科慧康”云平台的整体架构以及服务优化. 从平台整体架构设计和系统优化入手, 从负载均衡, 分布式缓存, 数据库优化, socket 转发服务器等方面设计了良好的云平台以支撑业务发展. 通过实际实验, 在系统负载稳定的情况下, 可以转发百万级别的数据量. 并且通过 http_load 对网站测试, 网页响应速度快, 后台系统稳定. 应用实践表明了该方案的可行性.

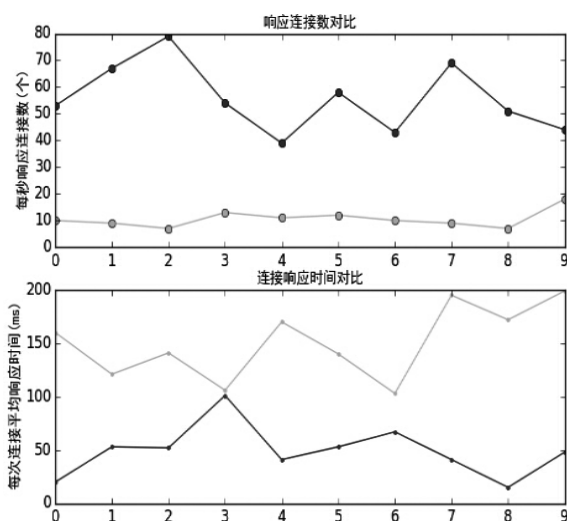


图 11 优化前后的关键参数对比

参考文献

- 1 刘翰腾,周毅,李小华,王毅,胡珊.移动医疗信息集成平台的构建.中国数字医学,2014,(4).
- 2 顾宏明.基于云平台的移动医疗健康服务系统的设计与实现[硕士学位论文].北京:北京邮电大学,2012.
- 3 林昊.大型网站架构演变和知识体系.程序员,2008,(11):66-69
- 4 王木年,曹先彬.Web 缓存替换策略与预取技术的研究[硕士学位论文].中国科学技术大学,2011.
- 5 曾庆江.负载均衡系统设计方案.计算机工程与设计,2009,(10).
- 6 覃伟,何新华.基于高可用性负载均衡的 Web 票务平台研究.计算机工程与设计,2009,(3).
- 7 包立辉,黄彦飞.高并发网站的架构研究及解决方案.计算机科学,2012,(10).
- 8 朱明纶.Linux 下基于 B_S 的高并发 Web 服务优化研究[硕士学位论文].上海:上海师范大学,2006.
- 9 陈亮.集群负载均衡关键技术研究[硕士学位论文].长沙:中南大学,2009.
- 10 李军峰,何明琪.高并发 Web 航空票务秒杀系统的设计与实现.计算机工程与设计,2013.
- 11 俞华锋.Memcached 在大型网站中的应用.科技信息,2008,(1).
- 12 Bakalova R, Kodali N, ChowShupp A. Web here dynamic cache: Improving J2EE application. IBM Systems Journal. July 2004, (20): 351.
- 13 浦云明,王宝玉.基于负载性能指标的 Web 测试.计算机系统应用,2010,19(5):220-223.