

# 基于 J2EE 技术的健身网站设计及实现<sup>①</sup>

## Design and Implementation of Get Fit Website Based on J2EE Technology

熊继平 (浙江师范大学 数理与信息工程学院通信工程系 浙江 金华 321004)

**摘 要:** 本文介绍设计的健身网站的整体框架及实现。整个系统选用 Apache 开源服务器作为集群的前端,负责接收来自客户的请求,以 MOD\_JK 的方式调度到多个中间服务器,实现大流量下的均衡调度;选用开源的 JBoss 和 Tomcat 作为 J2EE 以及 JSP 容器,分别支撑业务逻辑和页面逻辑。本文采用源代码集群结构系统对于类似的网站设计有着较好的借鉴和参考意义。此外,独特的针对不同用户能够自动创建个性化的合理健身计划模块是本文的创新点,也是网站商业化的支撑。

**关键词:** J2EE Apache 开源 集群 JNDI 健身计划

目前,各式各样的网站层出不穷,但大部分是门户网站以及内容管理系统(CMS)<sup>[1]</sup>,针对特定领域的专业网站依然比较少。鉴于此,本文将详细的介绍我们针对健身领域设计的智能健身计划定制网站<sup>[2]</sup>,该系统针对不同用户能够产生合理及有效的健身计划,能够满足从普通健身者到专业运动员的健身需求。此外,系统全部采用开源软件作为平台,构建了具有商业化性能的集群网站。

论文的剩余部分分成 4 个部分:第一部分首先介绍整个网站的集群结构;第二部分概要的介绍基于开源代码的业务逻辑实现和相关配置;第三部分介绍健身模块的核心思想;最后一部分总结全文。

### 1 具有可扩展性的集群结构

从将来发展的角度来考虑,为了应付大流量及并发性的服务需求,网站采用集群的结构。集群这也是如今大型网站必然的一种选择。

图 1 给出了本网站的集群结构。网站分为三个层次:Web 前端、中间层以及后端。

Web 前端为第一层,由 Apache2 服务器实现。从图中可见该服务器拥有外部和内部两种 IP 地址。这是出于安全考虑,只有 web 前端的服务器具有外

部 IP 地址,其余服务器只有内部地址(比如 192.168.\*.\* 之类的内部地址),这样有利于降低安全风险以及方便安装部署或升级第三方的安全解决方案,比如安装防火墙、IDS 或 IPS 之类的安全软件(只需要在直接对外的 Web 前端部署)。外部 IP 地址绑定 http 80 端口,负责接收所有对网站的访问请求,然后利用 MOD\_JK 技术针对不同的 url 请求分派到不同的中间层服务器。

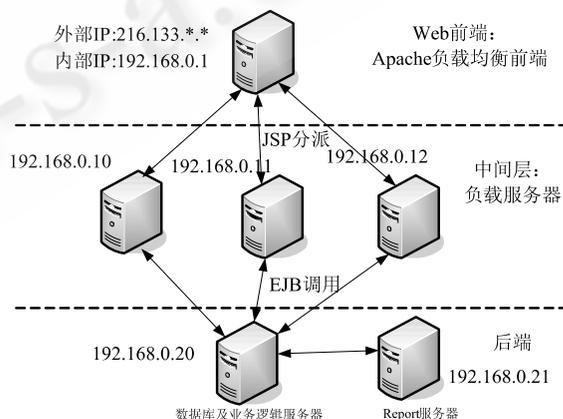


图 1 网站的集群结构

表 1 给出了实现该 MOD\_JK 技术所需的关键配置内容。其中第 1 行定义了集群的名称 cluster1,行 2-8

① 基金项目:浙江师范大学人才引进课题(ZC304006123)

定义了一个负载均衡服务器,也即我们系统中的中间服务器,第 9 行定义了集群的负载均衡方式,第 10 行给出了集群中的所有机器名称,这里只包含了一台机器。类似的,我们还可以定义集群 cluster2 和 cluster3 等。定义后集群后,我们只需要在 http.conf 文件中用 JkMount 命令将不同的 url 定位到不同的集群,从而实现了一种方式的负载均衡。从我们系统的实际商业化运行过程中可以看到,负载能够较为均匀的分配到中间服务器中,性能以及将来的可扩展性得到了充分保障。

表 1 MOD\_JK 的配置文件

```

1. worker.list = cluster1
# 定义其中的一台中间服务器
2. worker.tomcat1.port = 8009
3. worker.tomcat1.host = 192.168.0.11
4. worker.tomcat1.type = ajp13
5. worker.tomcat1.lbfactor = 100
6. worker.tomcat1.cachesize = 100
7. worker.tomcat1.cache_timeout = 600
8. worker.tomcat1.socket_timeout = 300
# 定义负载均衡方式
9. worker.cluster1.type = lb
10. worker.cluster1.balanced_workers = tomcat1
    
```

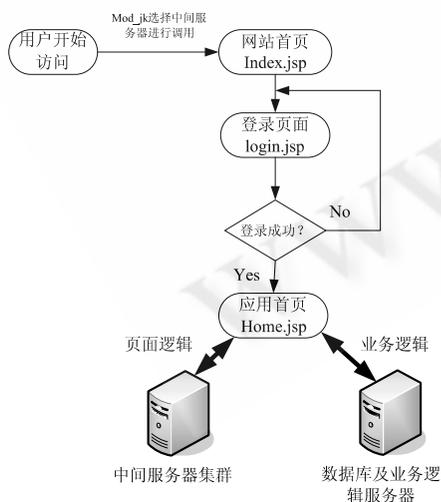


图 2 页面流程处理框图

## 2 J2EE 业务逻辑的实现

采用 J2EE 技术构造大型网站是目前的首选方案<sup>[5]</sup>,我们的系统也采用 J2EE 架构。J2EE 的业务逻辑也称商业逻辑,一般要实现和 JSP 页面逻辑的分离,也即需要实现远程调用方法,对于 Java 语言来说,选择 Sun 公司制定的 EJB (Enterprise Java Bean) 就是很自然的选择。同时,EJB 也是 J2EE 的核心,本质上是对数据库的一种远程调用能力的封装。我们选用目前被广泛使用的开源的 JBOSS [3] 作为 J2EE 容器,也即作为 EJB 的实现环境。事实上,经过少量修改网站也能够运行在商业 J2EE 容器之上,比如 WebLogic 等。关于 JBoss 的 EJB 详细配置过程可以参看文献<sup>[4]</sup>。

在网站的层次结构中,所有的中间服务器以及后台服务器都运行 JBoss + Tomcat 环境,其中 Tomcat 为 JSP 引擎。为了使得中间服务器能够调用 EJB,必须修改 JBoss 目录下的 jndi.properties 文件,给出运行 EJB 的数据库服务器的 IP 地址信息,类似于如下的修改 java.naming.provider.url = 192.168.0.20。

图 3 (a) 给出了 EJB 接口的定义,从中我们可以看到,网站的登录逻辑 LoginManager 是远程 EJB 接口定义。图 3 (b) 给出了具体实现的定义。采用类似的方式可以实现其它的业务逻辑,比如 ApplicationManager, AIManager 等。目前,我们按照功能功能划分,定义了 8 个业务逻辑。篇幅所限,就不一一介绍了。

表 2(a) EJB 的远程接口定义 (b) EJB 的具体实现

<pre> import java.rmi.*; import java.util.*; import javax.ejb.*; public interface LoginManager extends EJBObject, Remote, Manager { public LoginResultHelper login (String name, String password) throws RemoteException, InvalidLoginException; }         </pre>	<pre> public class LoginManagerBean implements SessionBean { //实现代码 }         </pre>
---	--

### 3 智能化的健身模块

图 3 给出了网站的主要部分,也即每个用户的健身计划 (programme)。健身计划的生成基于我们设计的能够针对不同用户的具有一定智能化的 AI 引擎。

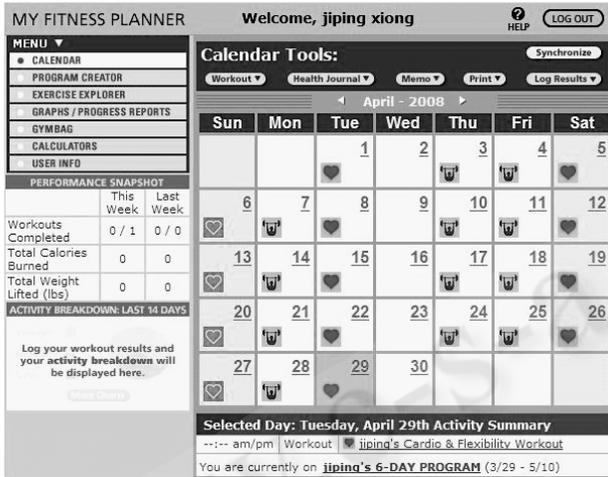


图 3 网站的主模块

首先我们会收集用户的个人健身信息,图 4 给出了某个用户填写的健身信息 (profile), 其中包括自身的数据、拥有的健身器材、每周的健身时间以及健身目标。

利用图 4 中所给出的这些信息以及用户历史训练情况,我们的 AI 引擎采用的 3 层并行产生、搜索以及优化机制,在所有大量的训练方案中搜寻最适合个人的健身方案。我们采用基于启发式的优化技术,让引擎能够在几分之一秒的时间内完成匹配任务,因此用户基本上不用等待,从而满足实时在线服务。流程框图如图 5 所示。

Personal Information	Time Constraints						
	Sun	Mon	Tue	Wed	Thu	Fri	Sat
<ul style="list-style-type: none"> <li>Age</li> <li>Gender</li> <li>Height, weight</li> <li>Body Measurements (journal)</li> <li>Fitness Levels (Strength, Cardio, Flexibility)</li> <li>Skill Levels (Strength, Cardio, Flexibility)</li> <li>Experience</li> <li>Health problems (PAR-Q)</li> <li>Injuries (muscles, joints) (PAR-Q)</li> </ul>	45 min	50 min	60 min	REST	60 min	35 min	REST
	Equipment Access			Goals List			
	Barbells	Yes	Upper-body strength Improve flexibility Improve posture Improve at sports: • Running • Tennis • Basketball • Karate				
	Dumbbells	Yes					
	Trap Bar	No					
	Hex Bar	Yes					
	Bicycle	Yes					
	Swimming Pool	No					
	Running Track	Yes					
	Treadmill	Yes					
	Ski Machine	No					

图 4 智能化训练系统中的用户相关信息输入

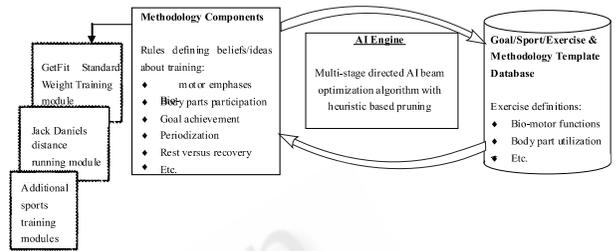


图 5 智能化的健身引擎框架

图 6 给出了智能引擎的输出。从中我们可以看到,输出是由训练 (workout) 构成一个多星期的训练计划,其中训练包含各种练习 (exercises) 和伸展操 (stretches)。并且每个练习都有制定的训练参数,例如有多少套,重复次数,体重,速度,时间,跑步机倾斜等等。

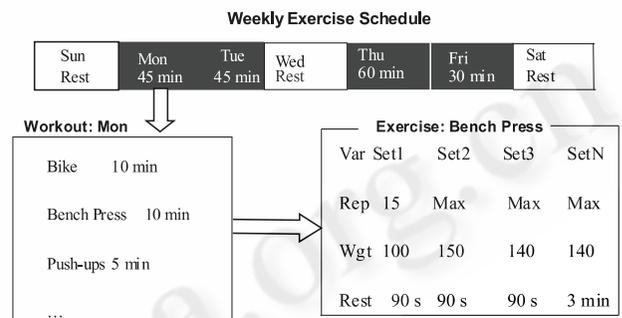


图 6 智能引擎输出的每周训练计划

### 4 结束语

本文从开发环境到核心模块,全面介绍了面向专门领域的健身网站系统。该系统已经对外开放,经过实际测试,在目前的集群条件下,能够支持数百人同时在线操作,具备商业化的条件。限于篇幅,该网站的其它功能未能涉及,比如基于 Worldpay 的在线支付模块等。本文所介绍的面向专门领域的网站开发思路对于其它类似的网站开发有着较好的示范和启发作用。相信随着国内支付手段的多样化,以及人们生活水平的提高,目前还不被重视的这种面向特定领域开发会得到越来越多的关注。

(下转第 75 页)

## 参考文献

- 1 Content Management System. [http://en.wikipedia.org/wiki/Content\\_management\\_system](http://en.wikipedia.org/wiki/Content_management_system)
- 2 Our Get Fit website, <http://83.138.132.68:8080/index.jsp>
- 3 JBoss official website <http://www.jboss.org>
- 4 JBoss3.0 下配置和部署 EJB 简介. [http://www.huihoo.org/jboss/jboss3\\_ejb.htm](http://www.huihoo.org/jboss/jboss3_ejb.htm)
- 5 陈军,段辉良. 基于 J2EE 的通用 Web 信息系统框架设计与实现. 计算机系统应用,2007,16(1)6-9.