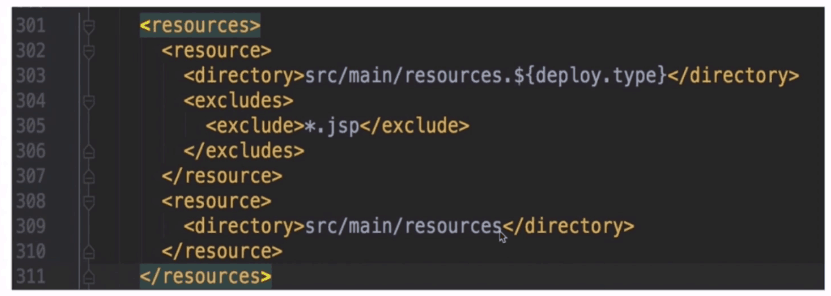
**1.本地项目使用cmd命令行打包，打包命令为：**

首先cd到项目所在目录，在执行命令

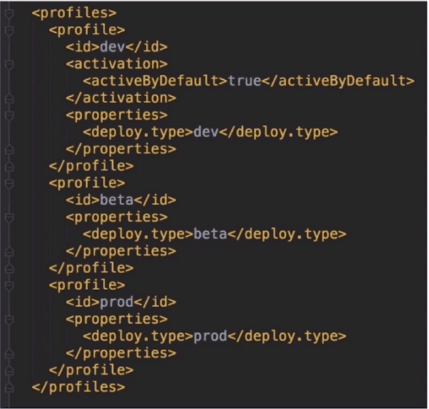
mvn clean package –Dmaven.test.skip=true (后面表示的是跳过测试)

**2.maven环境隔离配置**

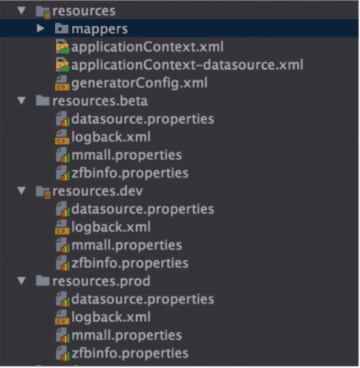
在pom.xml中build节点增加



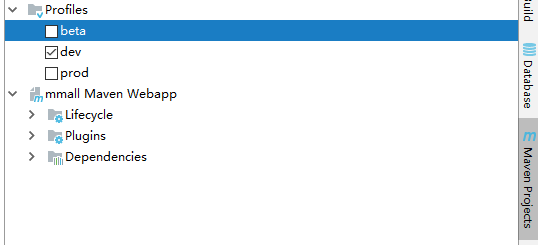
在pom.xml中增加profiles节点（profiles节点和build节点同级）



新建对应的文件夹，并把要隔离的文件分开，公共的留下



**3.maven环境隔离IDEA中设置默认环境**



**4.tomcat集群**

通过Nginx负载均衡进行请求转发。

Tomcat集群的问题：

（1）Session登录信息存储及读取的问题

（2）服务器定时任务并发的问题

解决方案：

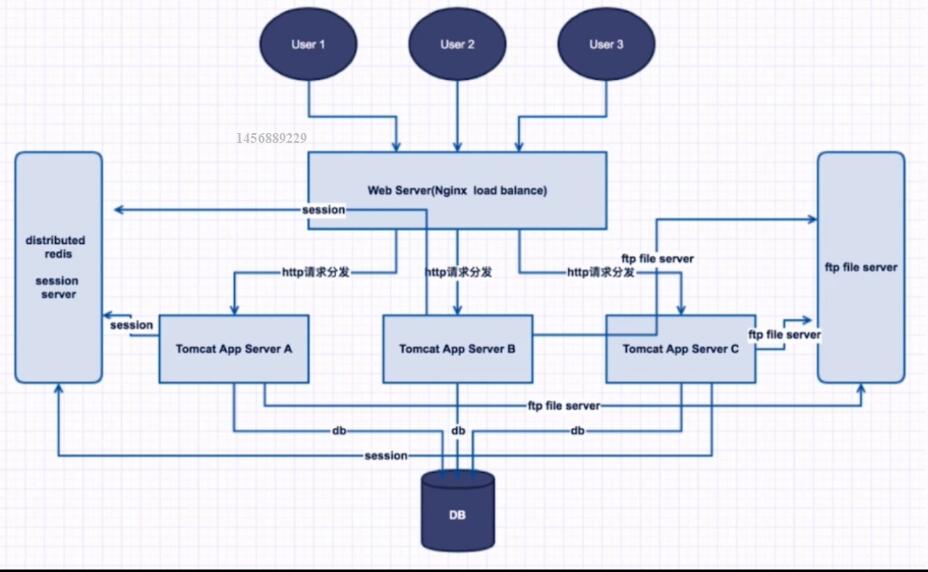
1. 采用nginx ip hash policy

优点：可以不改变现有技术架构，直接实现横向扩展（省事）

缺点：1. 导致服务器请求（负载）不平均（完全依赖ip hash）的结果

2. 在IP变化的环境下无法服务

二期真架构



**5.Nginx负载均衡配置。常用策略。**

① 轮询

优点：实现简单

缺点：不考虑每台服务器处理能力



1. 权重

优点：考虑了每台服务器处理能力的不同



1. ip hash

优点：能实现同一个用户访问同一个服务器

缺点：根据ip hash不一定平均



④url hash（第三方）

优点：能实现同一个用户访问同一个服务器

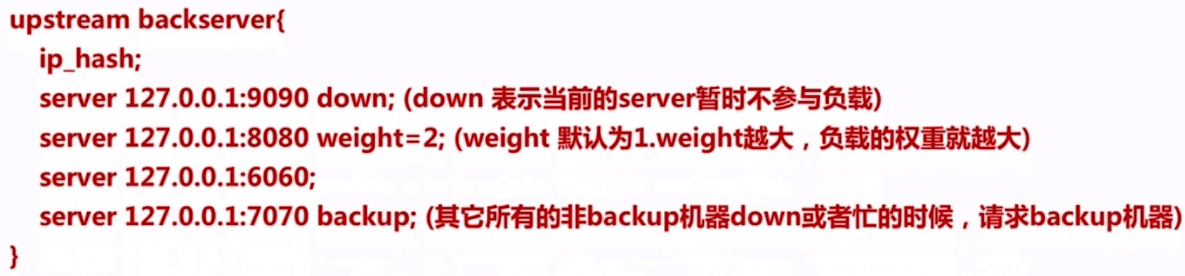
缺点：根据url hash分配请求会不平均，请求频繁的url会请求到同一个服务器上



⑤fair（第三方）

优点：按后端服务器的响应时间来分配请求，响应时间短的优先分配

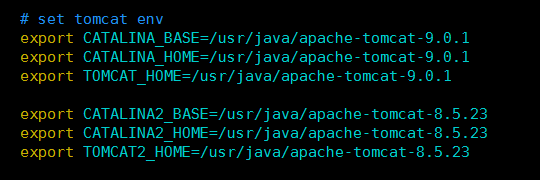




**Tomcat集群配置**

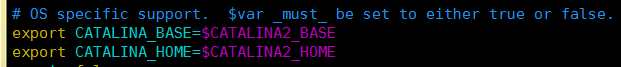
首先 /etc/profile增加tomcat环境变量

Tomcat 解压后

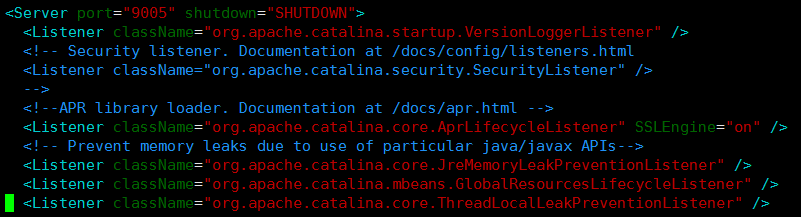


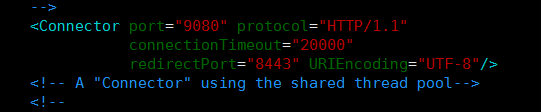
并要source /etc/profile使配置文件生效

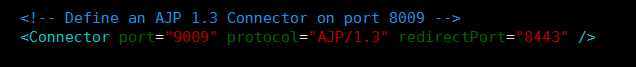
打开第二个tomcat目录bin下catalina.sh并找到 #OS specific support. $var\_must\_be set to either true or false. 在这行下面编辑（我的第二个tomcat是8.5.23）



打开第二个tomcat conf文件下的server.xml，修改三个地方的端口。



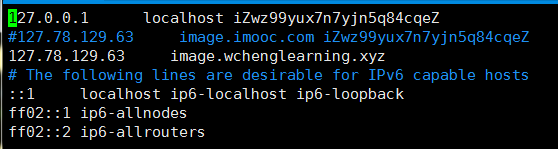


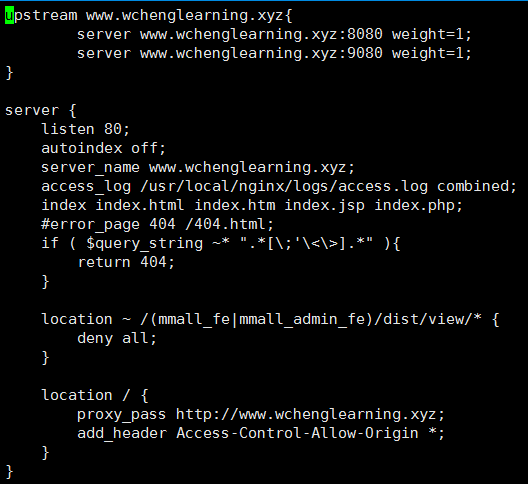


**Nginx + tomcat搭载集群配置：**

在Nginx下的conf文件下新建一个loadBalance.conf文件，编辑配置即可

首先cd /etc 编辑 vim hosts文件。添加自己的域名解析，然后配置负载均衡





**Redis的安装**

**1.linux下的安装**

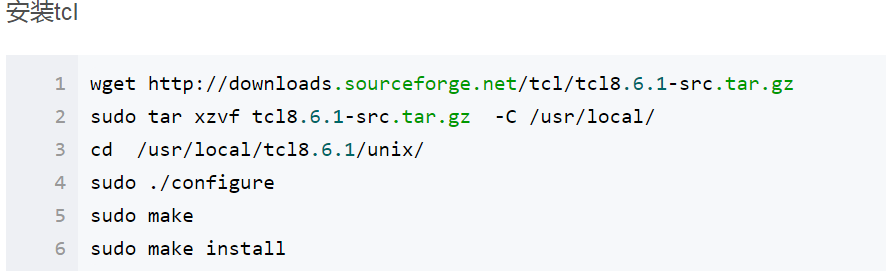
首先下载redis的压缩包：<http://download.redis.io/releases/>

解压到相应的文件夹 tar –zxvf redis-2.8.24 –C /usr

然后 cd /usr/redis-2.8.24 输入命令make

make完后会生成一个src目录，这时测试有没有编译成功，输入命令make test

如果出现报错You need tcl 8.5 or newer in order to run the Redis test，则按照如下步骤操作：



安装成功后启动redis ： ./redis-server 这时redis等待连接，新开一个窗口进入redis/src

输入./redis-cli就可连接成功。但是此时窗口是被占用的。按下Ctrl+C即可退出窗口占用。让redis后台启动 输入命令 ./redis-server &

Redis的关闭命令: ./redis-cli shutdown

2.windows下的安装

因为redis官方没有提供windows版本的，windows版本都是由微软团队进行维护的。所以，先下载redis windows版本：<https://github.com/MicrosoftArchive/redis/releases>

下载完解压缩即可用命令和linux下一样

Linux下redis启动：

默认启动 ./redis-server

默认连接 ./redis-cli

默认关闭 ./redis-cli shutdown

在指定端口启动 ./redis-server --port 端口号

在指定端口连接 ./redis-cli –p 端口号

在指定端口关闭 ./redis-cli –p 端口号

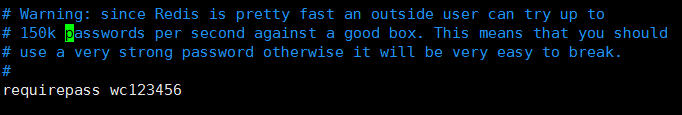
通过修改配置文件启动：./redis-server ../redis.conf

指定IP连接：./redis-cli –p 6379 –h 127.0.0.1 （如果远程连接的话，可以把ip改成远程ip）

指定ip关闭：./redis-cli –p 6379 –h 127.0.0.1 shutdown

配置redis密码：

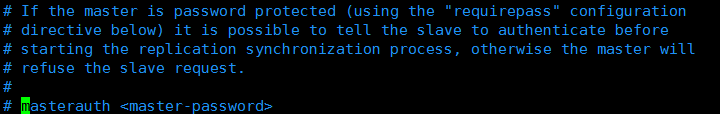
在redis.conf文件夹下找到requirepass 后面即为自己的密码



启动时就要用配置文件方式启动：./redis-server ../redis.conf

连接时就要用密码连接：./redis-cli –p 端口号 –a wc123456

当使用分布式时，有主从服务器，需要主从服务器的密码一致。先注释requirepass，找到masterauth 后面的即为自己的密码



Windows下的这些启动方式一样