**淘淘商城**

**第十四天**

# 课程计划

1. 发布前的准备工作
2. 了解系统部署架构
3. Nginx集群测试
4. MyCat(MySQL集群）

# 发布前的准备工作

## 准备工作

1. 确认服务器资源。
   1. 硬件资源：cpu、内存、硬盘
   2. 软件资源：window，Linux，是32位 64位
   3. 网络资源：是否需要在一个机房，百兆网络，千兆网络
2. 最终测试
   1. 开发人员，测试人员等全员参与
3. 需要在svn上打tag

在svn上打包发布版本

1. Maven打war包

Maven打包命令：package

1. 把数据库的结构和数据进行导出，做sql脚本
   1. 需要导出数据表和初始数据。
   2. 相关的资源也进行打包
2. 需要编写发布文档
   1. 发布文档应该在发布前两天就开始编写
   2. 发布文档编写全员参与
3. 准备回滚方案

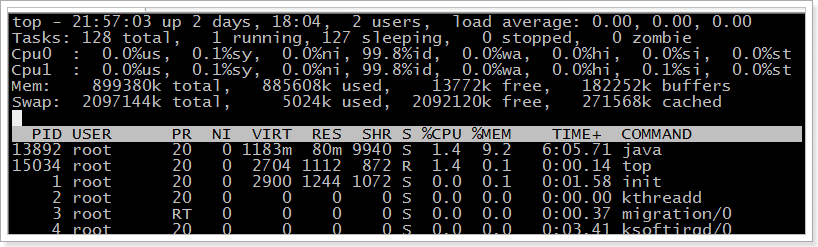
服务器初始化

## 确认服务器资源

### 确认cpu

[root@itcast-01 ~]# top

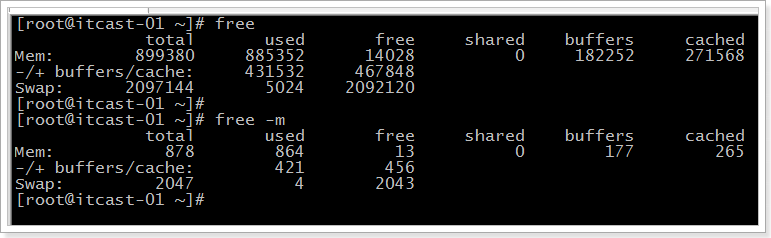
进入显示页面再按1，查看几核以及cpu使用情况



### 确认内存

[root@itcast-01 ~]# free -m

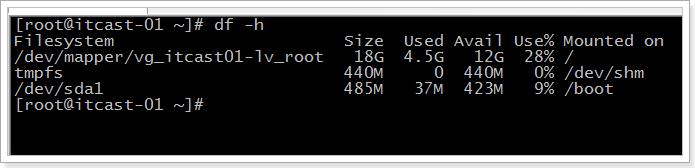
free 或者 free -m（表示以m为单位）



### 确认硬盘

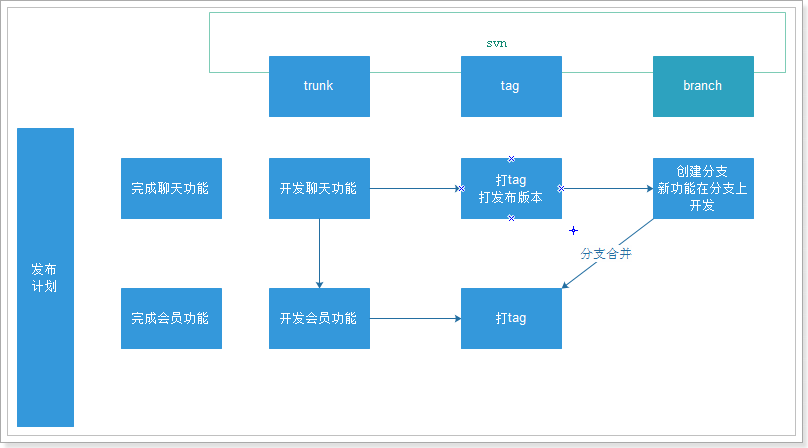
确认硬盘信息

[root@itcast-01 ~]# df -h



## 需要在svn上打tag

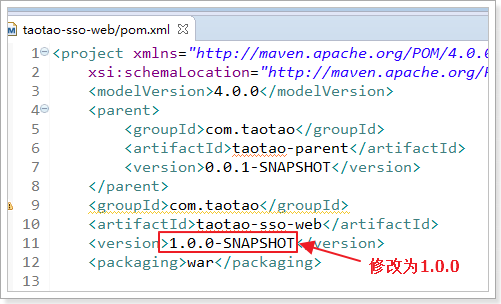
### trunk、tag、branch的作用



### svn打tag

#### 修改版本号

在需要打包的工程的pom.xml中，修改版本号



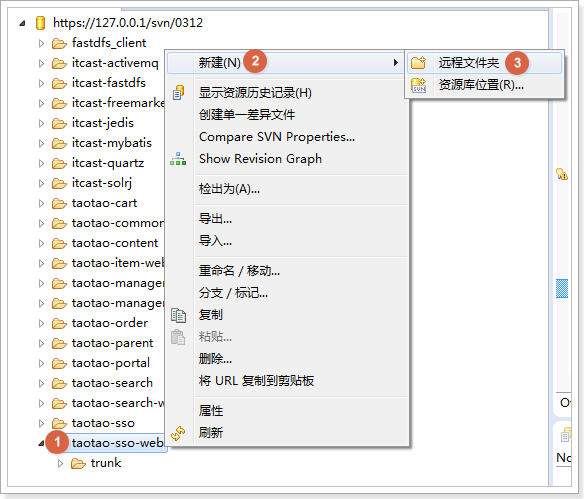
#### 在svn中打tag

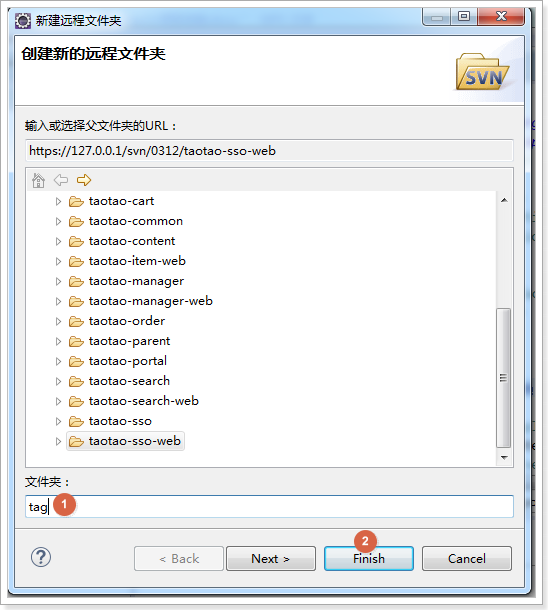
有两个方式：

1. 直接复制
2. 使用命令菜单

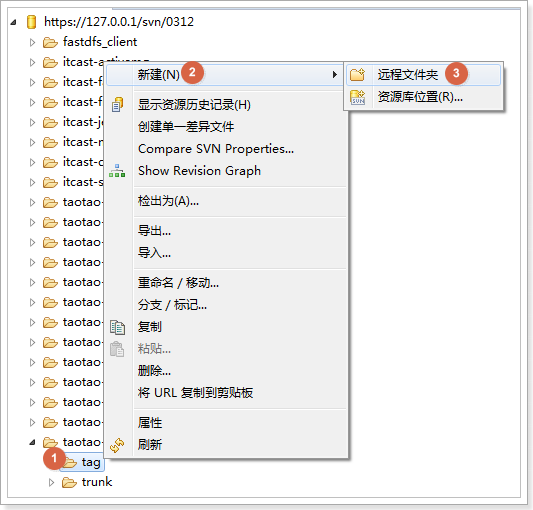
##### 直接复制

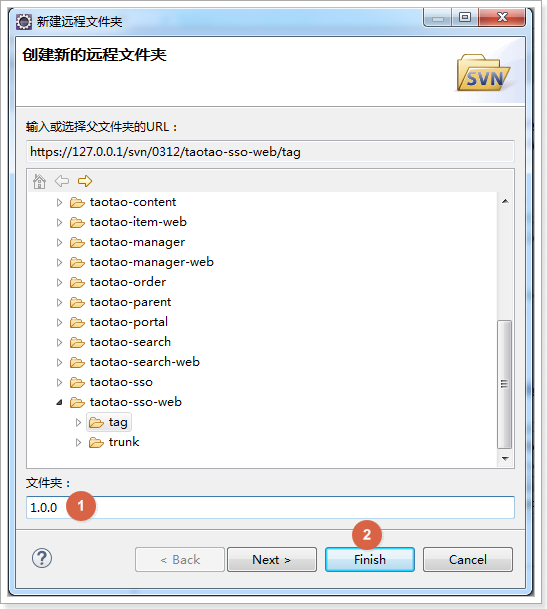
新建文件夹，起名叫tag



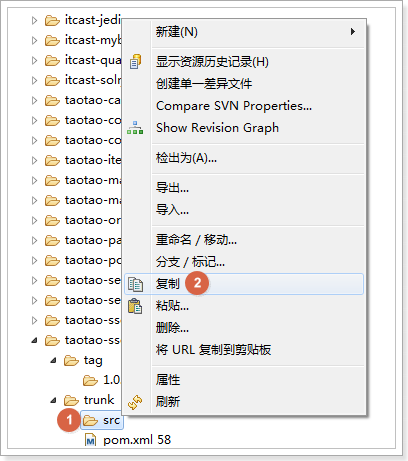


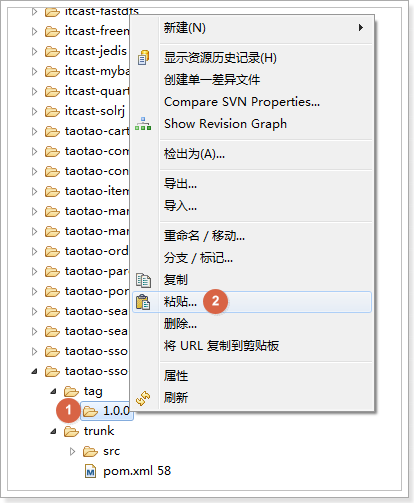
在其内创建文件夹，起名是需要打包工程的版本号

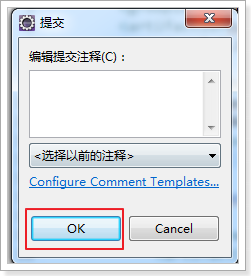




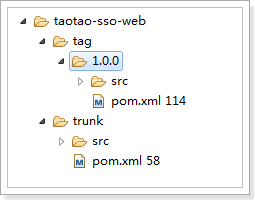
复制工程代码到tag/1.0.0中,需要复制src和pom.xml







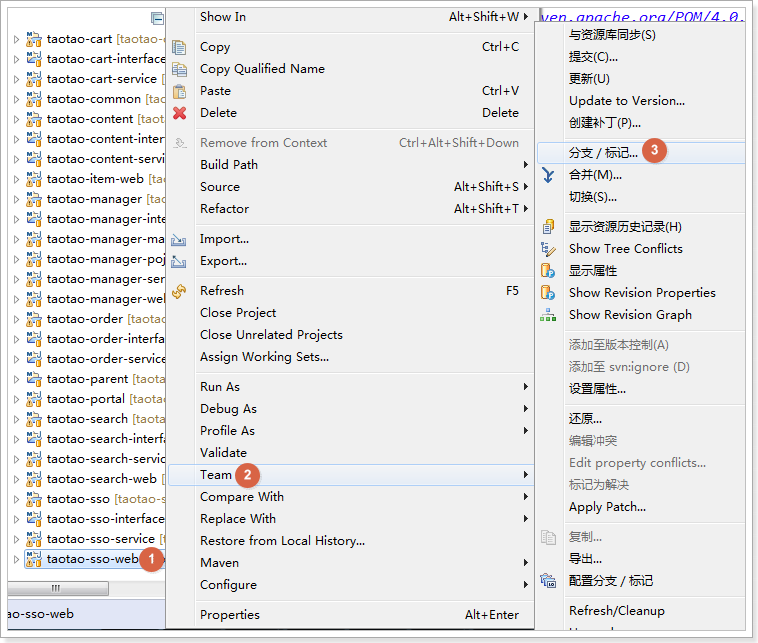
效果

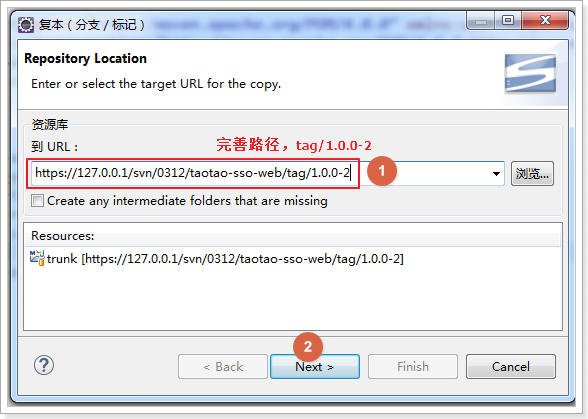


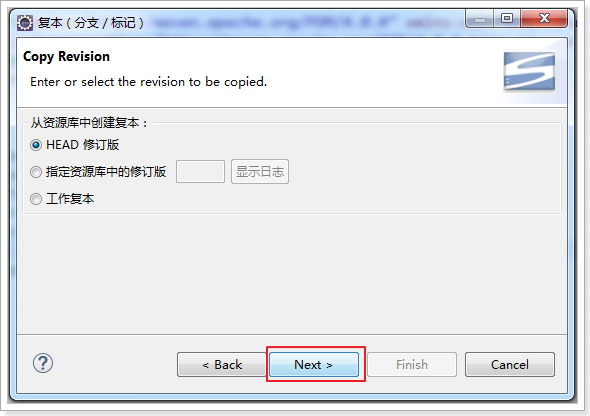
##### 使用命令菜单

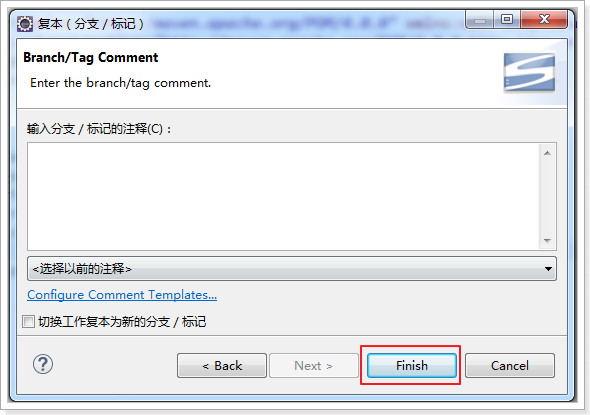
在java视图

工程上点击右键，创建标签

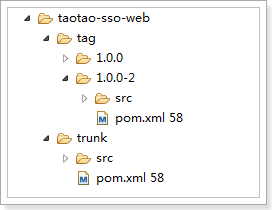






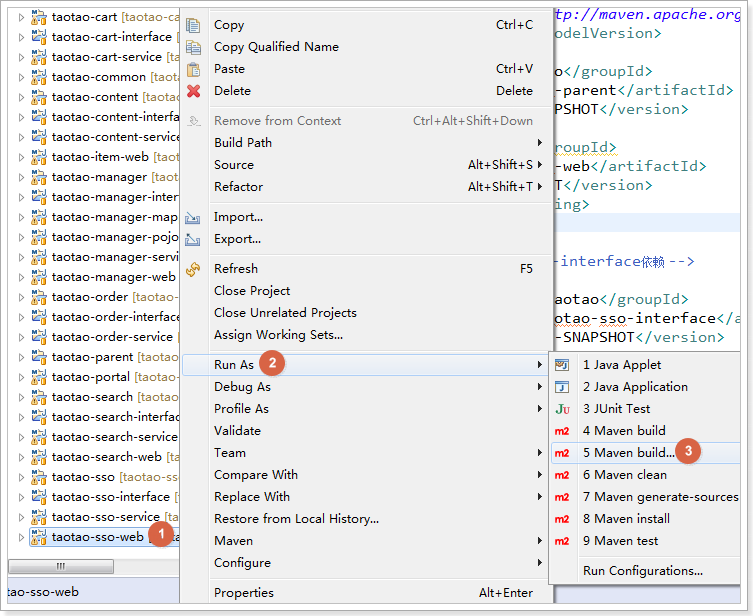


效果



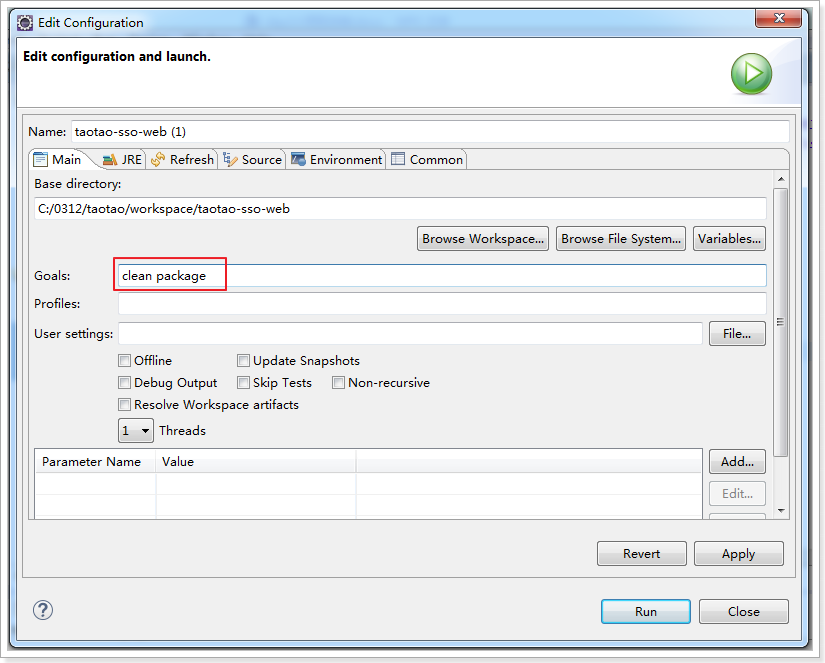
## Maven打war包

选择Maven build...



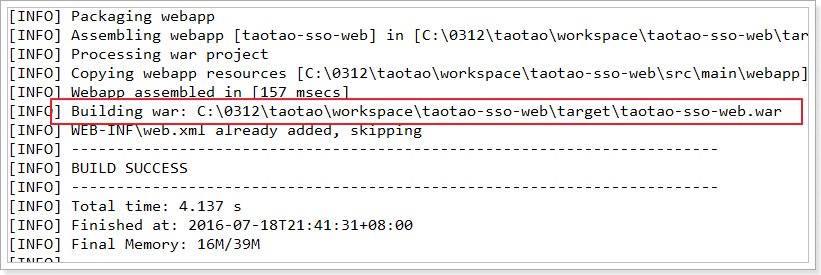
输入命令package

打包前最好清理一下clean



打包成功

打包所在位置

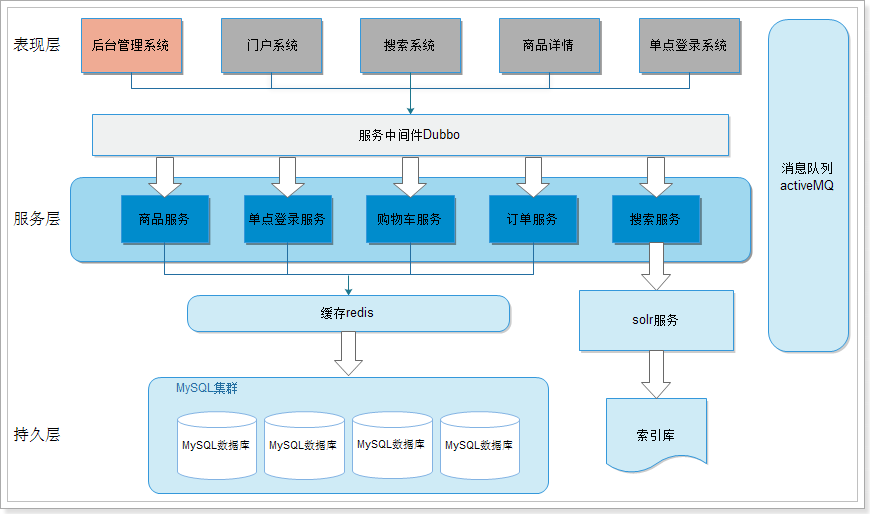


# 了解系统架构

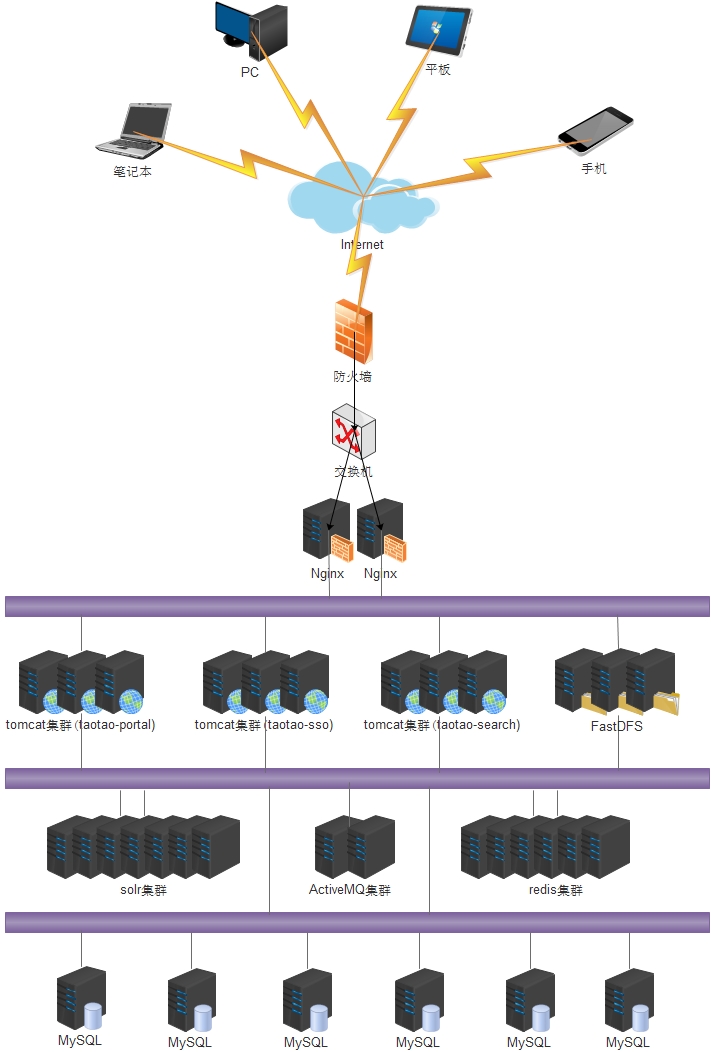
## 系统功能图



## 淘淘商城系统架构



## 网络拓扑图



## 系统部署

taotao-manager 2

taotao-manager-web 2

taotao-portal 4

taotao-sso 2

taotao-sso-web 2

taotao-search 2

taotao-search-web 2

taotao-item-web 2

taotao-cart 2

taotao-order 2

需要22台服务器。 22

Mysql 2

Solr 7

Redis 6

图片服务器 6

Nginx 2

注册中心 3

Activemq 2

共需要50台服务器。

### 域名规划

|  |  |  |
| --- | --- | --- |
| 序号 | 工程名 | 域名 |
| 1 | taotao-manager-web | manager.taotao.com |
| 2 | taotao-portal | www.taotao.com |
| 3 | taotao-search-web | search.taotao.com |
| 4 | taotao-item-web | item.taotao.com |
| 5 | taotao-sso-web | sso.taotao.com |

只用申请一个一级域名即可

www.taotao.com taotao.com

# Nginx集群测试

## 安装Nginx

### 环境准备

克隆一个新的Linux



安装gcc-c++

Nginx是c语言开发的，我们使用源码安装，需要先进行编译，在安装，所以需要gcc-c++库

[root@itcast-01 ~]# yum install gcc-c++

### 安装Nginx

#### 安装环境：

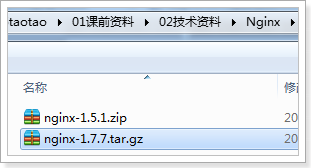
安装pcre库

yum -y install pcre-devel

安装zlib库

yum install -y zlib-devel

#### 上传Nginx



按alt+p进入上传界面，上传Nginx

#### 解压

解压

[root@itcast-01 ~]# tar -zxvf nginx-1.7.7.tar.gz

进入解压文件夹

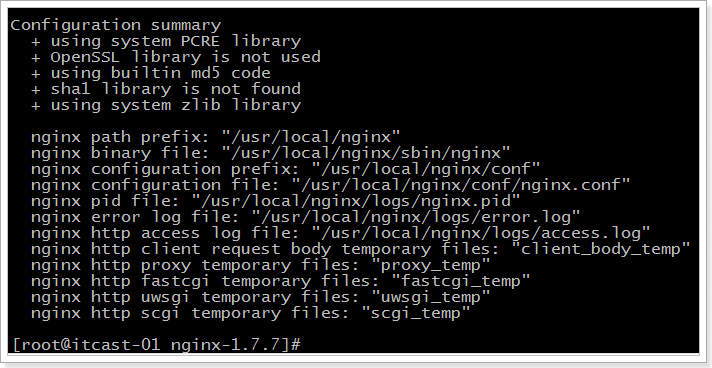
[root@itcast-01 ~]# cd nginx-1.7.7

#### 编译安装

设置安装参数

[root@itcast-01 nginx-1.7.7]# ./configure --prefix=/usr/local/nginx

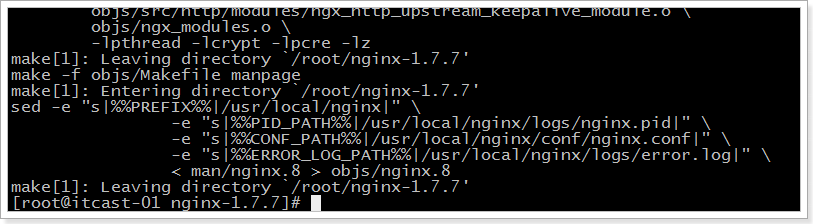
设置成功效果



编译并安装

[root@itcast-01 nginx-1.7.7]# make

编译成功效果

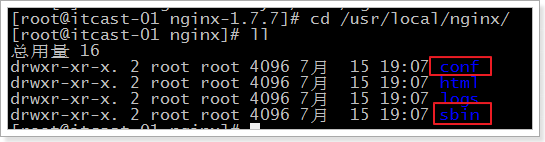


[root@itcast-01 nginx-1.7.7]# make install

#### 启动Nginx

查看安装文件，conf是配置文件，sbin是启动目录

[root@itcast-01 nginx-1.7.7]# cd /usr/local/nginx/

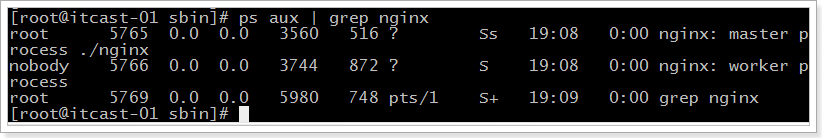


进入启动文件目录，启动Nginx

[root@itcast-01 nginx]# cd sbin/

[root@itcast-01 sbin]# ./nginx

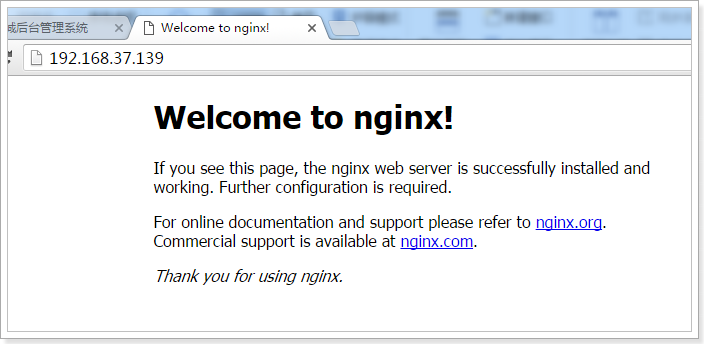
查看启动进程



关闭防火墙

[root@itcast-01 sbin]# service iptables stop

访问测试

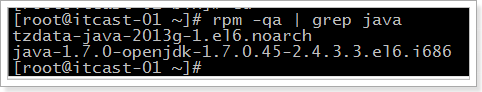


## 安装jdk

### 删除原有的jdk

1. 查看原有的jdk

[root@itcast-01 ~]# rpm -qa | grep java

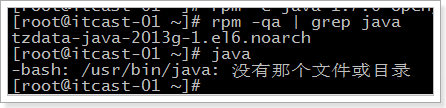


发现已安装openjdk，需要卸载

1. 删除原有的jdk

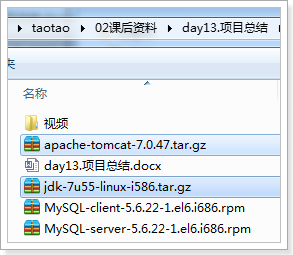
[root@itcast-01 ~]# rpm -e java-1.7.0-openjdk-1.7.0.45-2.4.3.3.el6.i686 --nodeps

再次查看，并执行java命令，确认openjdk已经删除



### 安装JDK

1. jdk准备



2. 上传jdk

按alt+p进入SFTP界面，进行上传

1. 解压jdk

[root@itcast-01 ~]# tar -zxvf jdk-7u55-linux-i586.tar.gz

1. 移动jdk

[root@itcast-01 ~]# mv jdk1.7.0\_55/ /usr/local/

1. 配置环境变量

[root@itcast-01 ~]# vim /etc/profile

在最下方添加如下两行配置

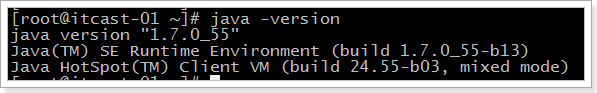
export JAVA\_HOME=/usr/local/jdk1.7.0\_55

export PATH=$JAVA\_HOME/bin:$PATH

1. 设置环境变量生效

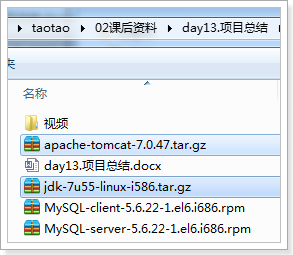
[root@itcast-01 ~]# source /etc/profile

输入命令测试java配置是否生效



## 安装Tomcat

Tomcat准备



1. 上传

按alt+p进入SFTP界面，进行上传

1. 解压

[root@itcast-01 ~]# tar -zxvf apache-tomcat-7.0.47.tar.gz

1. 移动

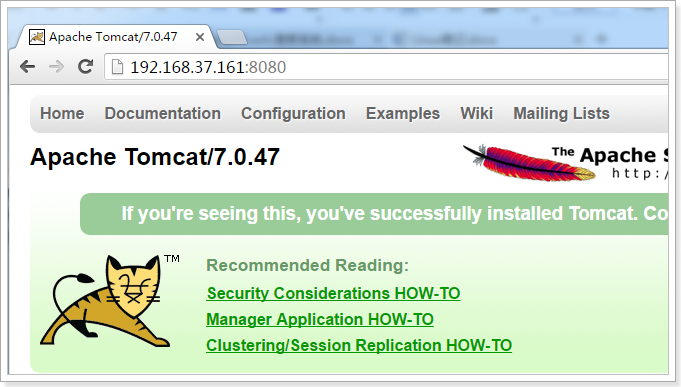
[root@itcast-01 ~]# mkdir /usr/local/web

[root@itcast-01 ~]# mv apache-tomcat-7.0.47 /usr/local/web/tomcat

1. 启动测试

[root@itcast-01 ~]# /usr/local/web/tomcat/bin/startup.sh

测试：

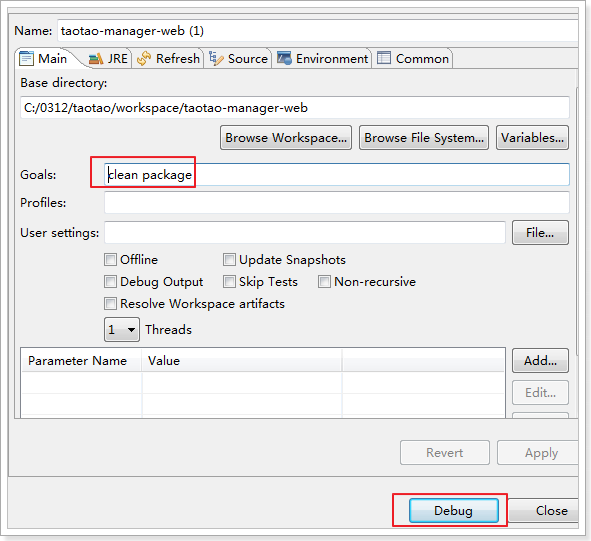


## 部署项目

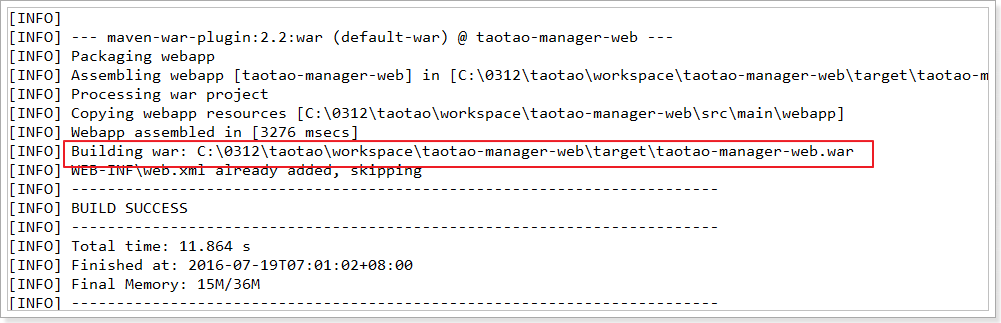
这里使用后台系统进行测试

搭建两个taotao-manager-web，作为集群

### 给taotao-manager-web打war包



效果



### 上传到Linux

alt+p上传

## Tomcat准备

在Linux复制一份Tomcat，并清理webapps

进入文件夹

[root@itcast-01 ~]# cd /usr/local/web/

停止Tomcat

[root@itcast-01 web]# tomcat/bin/shutdown.sh

复制一份Tomcat

[root@itcast-01 web]# cp -r tomcat/ tomcat-manager-web1

删除webapps的内容，创建ROOT文件夹

[root@itcast-01 web]# cd tomcat-manager-web1/webapps/

[root@itcast-01 webapps]# rm -rf \*

[root@itcast-01 webapps]# mkdir ROOT

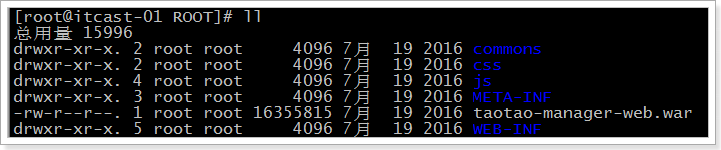
把上传的自己的taotao-manager-web，移动到ROOT中

[root@itcast-01 webapps]# mv /root/taotao-manager-web.war ROOT/

解压war包

[root@itcast-01 webapps]# cd ROOT/

[root@itcast-01 ROOT]# jar -xvf taotao-manager-web.war



删除war包

[root@itcast-01 ROOT]# rm -f taotao-manager-web.war

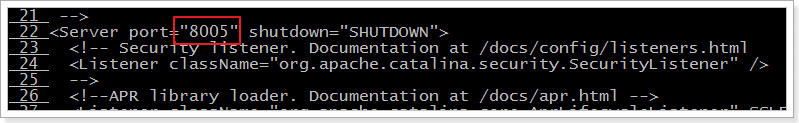
需要搭建集群，复制一份配置好的Tomcat

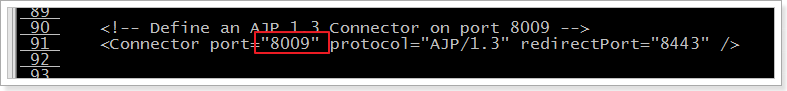
[root@itcast-01 web]# cp -r tomcat-manager-web1/ tomcat-manager-web2

修改两个Tomcat的端口号，分别改为8081和8082

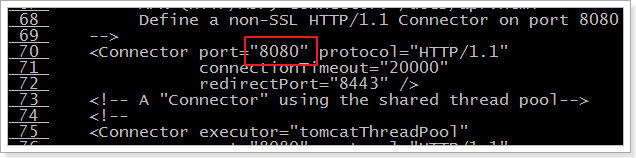
[root@itcast-01 web]# vim tomcat-manager-web1/conf/server.xml

需要修改3个地方，8005和8009的地方随意修改，不要使用别人占用的端口号即可





8080的位置修改为8081和8082



启动两个Tomcat

[root@itcast-01 web]# tomcat-manager-web1/bin/startup.sh

[root@itcast-01 web]# tomcat-manager-web2/bin/startup.sh

## 配置Nginx集群

安装Nginx

上传解压，进入Nginx解压目录

执行

编译并安装

[root@itcast-01 nginx-1.7.7]# ./configure --prefix=/usr/local/nginx

[root@itcast-01 nginx-1.7.7]# make

[root@itcast-01 nginx-1.7.7]# make install

### 修改Nginx的nginx.conf文件

在http节点添加：

#配置集群信息

upstream taotao-manager-web {

server 192.168.37.136:8080;

server 192.168.37.136:8081;

}

server {

listen 80;

server\_name manager.taotao.com;

proxy\_set\_header X-Forwarded-Host $host;

proxy\_set\_header X-Forwarded-Server $host;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

location / {

#使用前面配置的集群

proxy\_pass http://taotao-manager-web;

proxy\_connect\_timeout 600;

proxy\_read\_timeout 600;

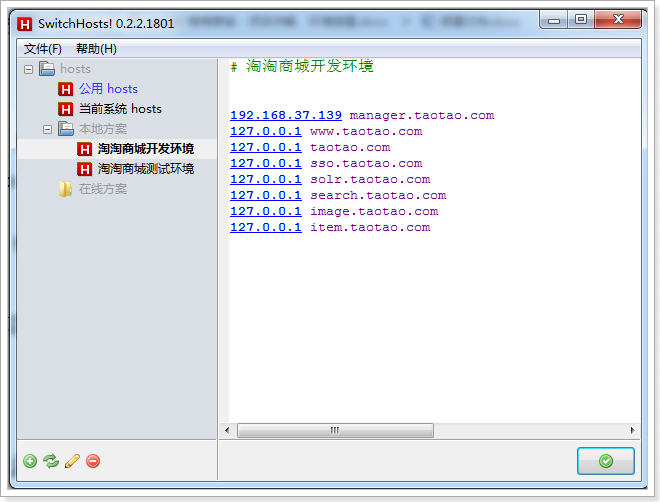
}

}

### 修改完成后重载Nginx

[root@itcast-01 sbin]# ./nginx -s reload

### 修改宿主机的hosts



# MyCat

## MyCat介绍

### 什么是MyCat？

简单的说，MyCat就是：

一个彻底开源的，面向企业应用开发的“大数据库集群”

支持事务、ACID、可以替代Mysql的加强版数据库

一个可以视为“Mysql”集群的企业级数据库，用来替代昂贵的Oracle集群

一个融合内存缓存技术、Nosql技术、HDFS大数据的新型SQL Server

结合传统数据库和新型分布式数据仓库的新一代企业级数据库产品

一个新颖的数据库中间件产品

MyCat的目标是：低成本的将现有的单机数据库和应用平滑迁移到“云”端，解决数据存储和业务规模迅速增长情况下的数据瓶颈问题。

### MyCat的关键特性

支持 SQL 92标准

支持Mysql集群，可以作为Proxy使用

支持JDBC连接ORACLE、DB2、SQL Server，将其模拟为MySQL Server使用

支持galera for mysql集群，percona-cluster或者mariadb cluster，提供高可用性数据分片集群

自动故障切换，高可用性

支持读写分离，支持Mysql双主多从，以及一主多从的模式

支持全局表，数据自动分片到多个节点，用于高效表关联查询

支持独有的基于E-R 关系的分片策略，实现了高效的表关联查询

多平台支持，部署和实施简单

### MyCat架构

**Client**

(MySQL CLI/JDBC/ODBC/…)

**MySQL Socket Protocol Handler**

**SQL Parser**

**SQL Router**

**Front**

**Server**

**Storage**

**SQL Executor**

**DataNode**

**HeartBeat Checker**

**MySQL instance A**

**MySQL instance B**

**MySQL instance C**

**More….**

如图所示：MyCat使用Mysql的通讯协议模拟成了一个Mysql服务器，并建立了完整的Schema（数据库）、Table （数据表）、User(用户)的逻辑模型，并将这套逻辑模型映射到后端的存储节点DataNode（MySQL Instance）上的真实物理库中，这样一来，所有能使用Mysql的客户端以及编程语言都能将MyCat当成是Mysql Server来使用，不必开发新的客户端协议。

MyCat对多数据库的支持



## 下载MyCat

官方网站：

[http://www.MyCat.org.cn/](http://www.mycat.org.cn/)

github地址

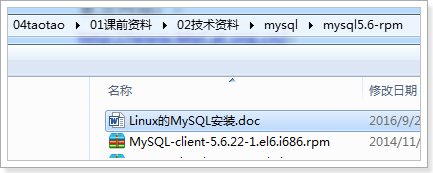
[https://github.com/MyCatApache](https://github.com/MyCATApache)

## 环境准备

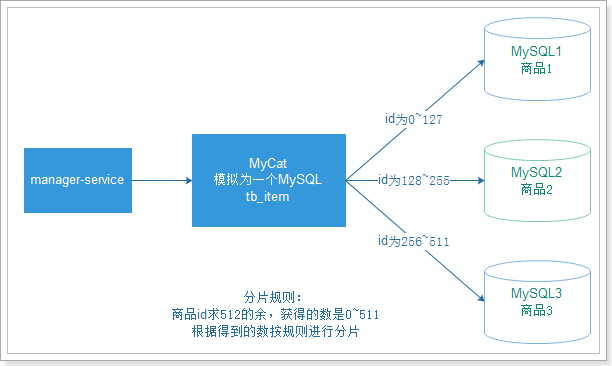
数据库准备,准备三个MySQL数据库。

在Linux上安装一个MySQL数据库进行模拟，在设置三个数据库，作为集群中的库进行使用。

Linux安装MySQL参考文档：

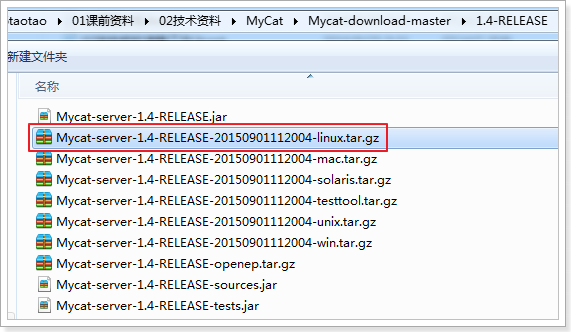


集群分片规则



## MyCat安装

1. 上传MyCat



1. 移动并解压MyCat

[root@itcast-01 ~]# mv MyCat-server-1.4-RELEASE-20150901112004-linux.tar.gz /usr/local/

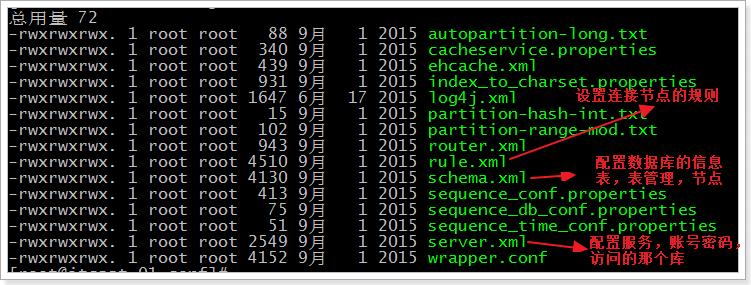
[root@itcast-01 ~]# cd /usr/local/

[root@itcast-01 local]# tar -zxf MyCat-server-1.4-RELEASE-20150901112004-linux.tar.gz

1. 配置MyCat

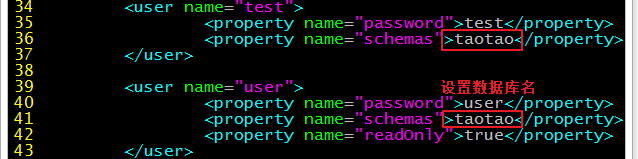
[root@itcast-01 local]# cd MyCat/

[root@itcast-01 MyCat]# cd conf/



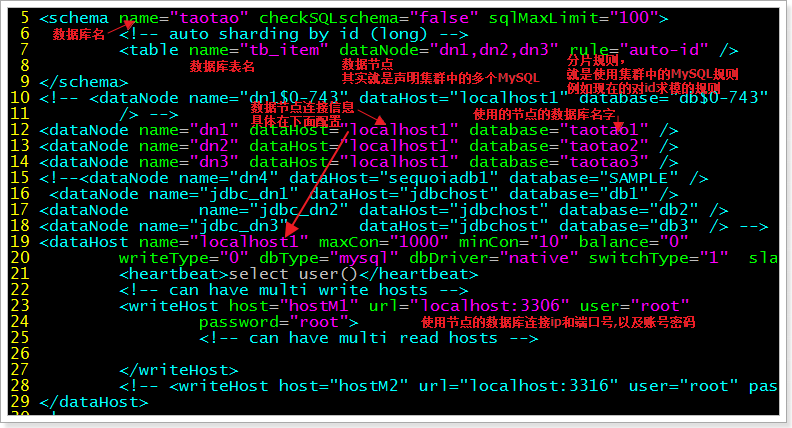
* 1. 配置server.xml

[root@itcast-01 conf]# vim server.xml



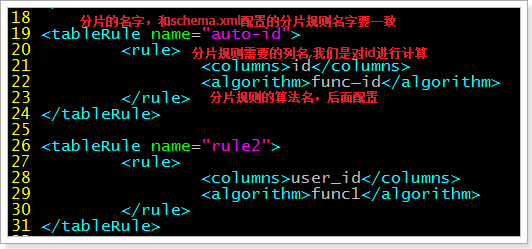
* 1. 配置schema.xml

[root@itcast-01 conf]# vim schema.xml



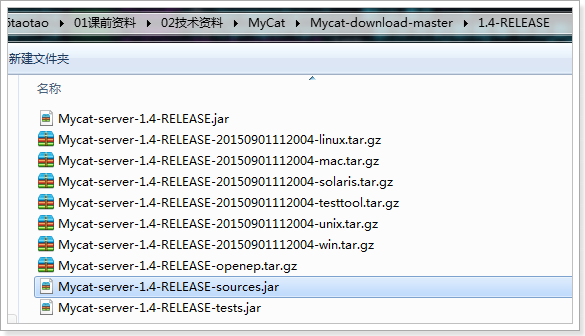
* 1. 配置rule.xml

[root@itcast-01 conf]# vim rule.xml



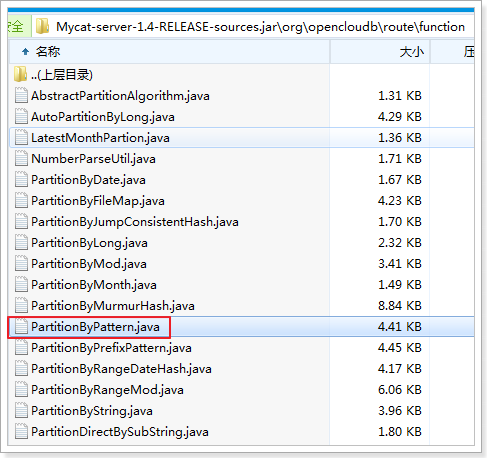
* 1. 查找分片算法

配置的分片规则算法，可以参考以下源码：



下面是压缩包中分片规则的算法，我们使用的是求余的方式：

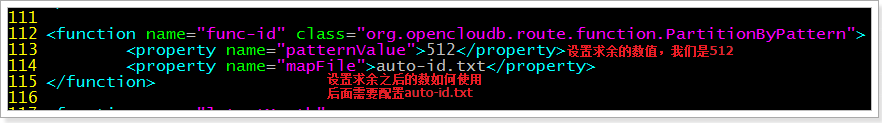
org.opencloudb.route.function.PartitionByPattern



* 1. 配置分片算法

根据分配片的java代码，继续在rule.xml配置分片算法

按下图方式配置分片规则算法：



* 1. 添加并编辑配置文件

[root@itcast-01 conf]# vim auto-id.txt

0-127=0

128-255=1

255-511=2

## 启动MyCat

[root@itcast-01 conf]# cd /usr/local/MyCat/bin/

[root@itcast-01 bin]# ./MyCat start

## MyCat测试

连接MyCat：

连接MyCat和链接MySQL是一样的，区别是，MyCat的端口号是8066

MyCat的账号密码在server.xml中进行配置

测试MyCat：

在MyCat中创建数据库表，发现使用的三个库都创建了表

在MyCat中插入数据，发现按照规则给三个数据库分别插入了数据