

淘淘商城 第十三天



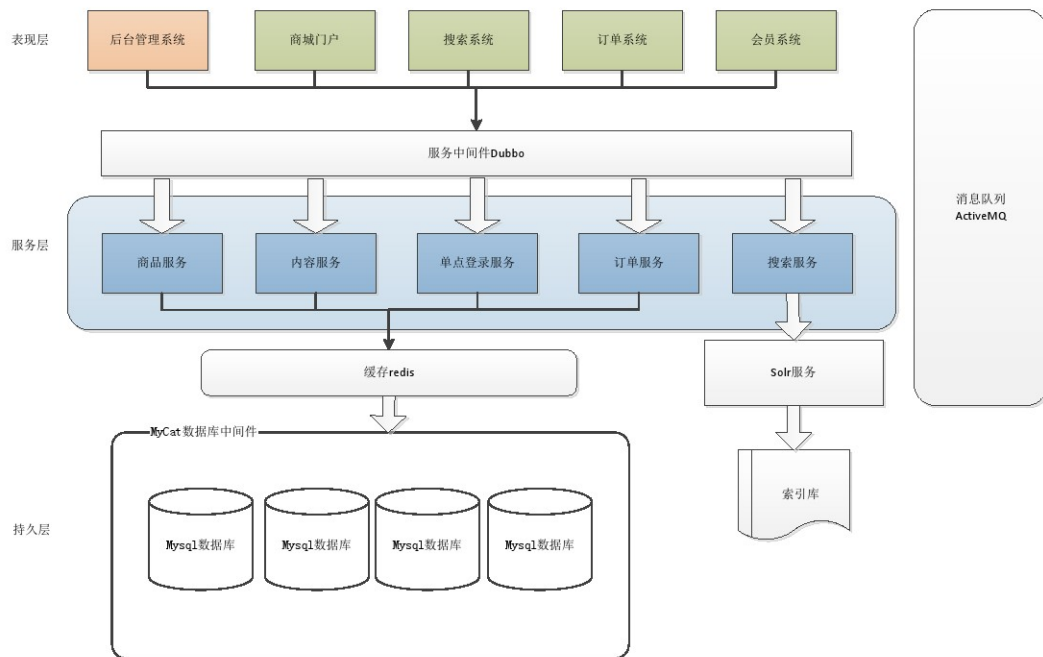
1. 课程计划

第十三天：

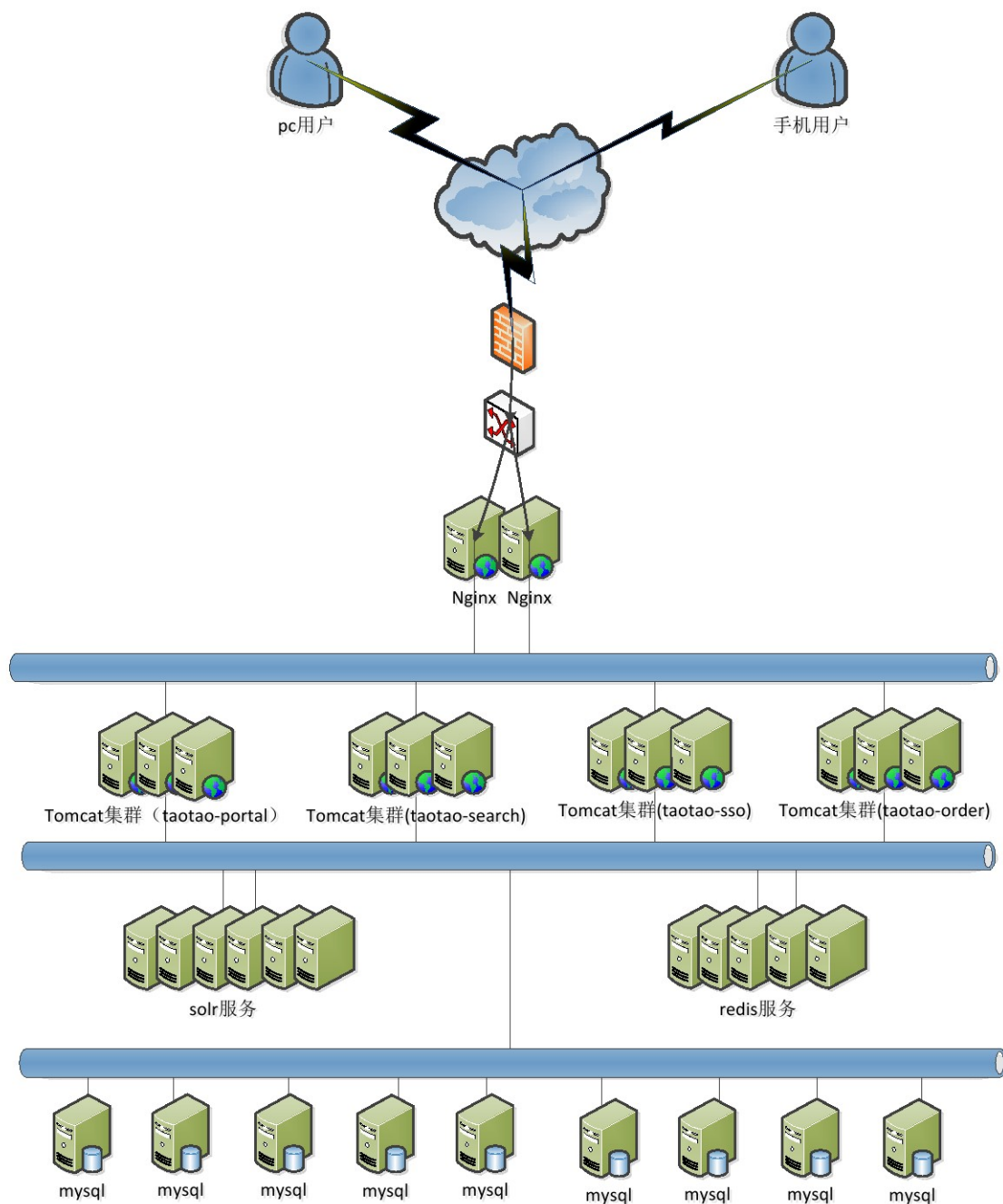
- 1、系统部署
- 2、项目总结
- 3、面试中的问题

2. 项目部署

2.1. 项目架构讲解



2.2. 网络拓扑图



2.3.Linux 下安装 mysql

第一步：查看 mysql 是否安装。

```
rpm -qa|grep mysql
```

第二步：如果 mysql 的版本不是想要的版本。需要把 mysql 卸载。

```
yum remove mysql mysql-server mysql-libs mysql-common
```

```
rm -rf /var/lib/mysql
```

```
rm /etc/my.cnf
```

第三步：安装 mysql。需要使用 yum 命令安装。在安装 mysql 之前需要安装 mysql 的下载源。需要从 oracle 的官方网站下载。

1) 下载 mysql 的源包。

我们是 centos6.4 对应的 rpm 包为: mysql-community-release-el6-5.noarch.rpm

2) 安装 mysql 下载源:

yum localinstall mysql-community-release-el6-5.noarch.rpm



3) 在线安装 mysql:

yum install mysql-community-server

第四步: 启动 mysql

service mysqld start

第五步: 需要给 root 用户设置密码。

/usr/bin/mysqladmin -u root password 'new-password' // 为 root 账号设置密码

第六步: 远程连接授权。

GRANT ALL PRIVILEGES ON *.* TO 'myuser'@'%' IDENTIFIED BY 'mypassword' WITH GRANT OPTION;

注意: 'myuser'、'mypassword' 需要替换成实际的用户名和密码。

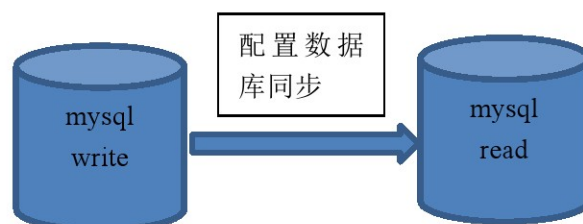
2.4. 数据库的读写分离

Mysql 提供的解决方案: 使用 binlog 进行数据库同步。需要配置 mysql。

代码中实现读写分类:

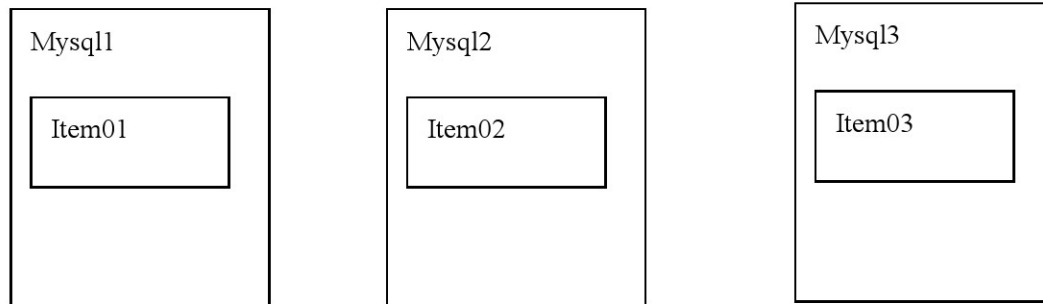
1、可以使用 aop 实现一个切面。动态切换数据源。需要编程实现。

2、使用数据库中间件实现读写分类, 分库分表。



分库分表：

当数据库的表中数据非常大的时候例如上千万条数据。查询性能非常低。可以把一张表保存到不同的数中。



可以使用一个数据库中间件 mycat。国产开源项目，前身是 cobar 项目。

2.5. 系统部署

2.5.1. 部署分析

Taotao-manager

Taotao-manager-web

Taotao-portal-web

Taotao-content

Taotao-search

Taotao-search-web

Taotao-item-web

Taotao-sso

Taotao-sso-web

Taotao-cart-web

Taotao-order

Taotao-order-web

需要 24 台服务器。 24

Mysql 2

Solr 7

Redis 6

图片服务器 2

Nginx 2

注册中心 3

Activemq 2

共需要 48 台服务器。

搭建伪分布式。

2.5.2. 服务器规划

项目	服务器数量	虚拟机	ip
Mysql	2	1	134
Solr	7	1	154
Redis	6	1	153
图片服务器	2	1	133
Nginx	2	1	141
注册中心	3	1	167
Activemq	2	1	168
Taotao-manager	8080		
Taotao-content	8081		
Taotao-search	8082		
~~~~~			1 135
Taotao-sso	8080		
Taotao-order	8081		
~~~~~			1 136
Taotao-manager-web	8080		
Taotao-portal-web	8081		
Taotao-search-web	8082		
~~~~~			1 137
Taotao-item-web	8080		
Taotao-sso-web	8081		
~~~~~			1 138
Taotao-cart-web	8080		
Taotao-order-web	8081		
~~~~~			1 139

## 2.5.3. 域名规划

序号	工程名	域名
1	Taotao-manager-web	manager.taotao.com
2	Taotao-portal-web	<a href="http://www.taotao.com">www.taotao.com</a>
3	Taotao-search-web	search.taotao.com
4	Taotao-item-web	item.taotao.com
5	Taotao-sso-web	sso.taotao.com
6	Taotao-cart-web	cart.taotao.com
7	Taotao-order-web	order.taotao.com

## 2.5.4. Tomcat 热部署





可以使用 maven 实现 tomcat 热部署。Tomcat 启动时 部署工程。

Tomcat 有个后台管理功能，可以实现工程热部署。

配置方法：

第一步：需要修改 tomcat 的 conf/tomcat-users.xml 配置文件。添加用户名、密码、权限。

```
<role rolename="manager-gui" />
<role rolename="manager-script" />
<user username="tomcat" password="tomcat" roles="manager-gui, manager-script"/>
```

第二步：重新启动 tomcat。

使用 maven 的 tomcat 插件实现热部署：

第一步：配置 tomcat 插件，需要修改工程的 pom 文件。

```
<build>
  <plugins>
    <!-- 配置 Tomcat 插件 -->
    <plugin>
      <groupId>org.apache.tomcat.maven</groupId>
      <artifactId>tomcat7-maven-plugin</artifactId>
      <configuration>
        <port>8081</port>
        <path>/</path>
        <url>http://192.168.25.135:8080/manager/text</url>
        <username>tomcat</username>
        <password>tomcat</password>
      </configuration>
    </plugin>
  </plugins>
</build>
```

第二步：使用 maven 命令进行部署。

tomcat7:deploy

tomcat7:redeploy

部署的路径是 “/” 会把系统部署到 webapps/ROOT 目录下。

部署工程跳过测试：

clean tomcat7:redeploy -DskipTests

## 2.5.5. 工程部署

每个工程运行在不同的 tomcat 上，修改 tomcat 的端口号。

## 2.6. 反向代理的配置

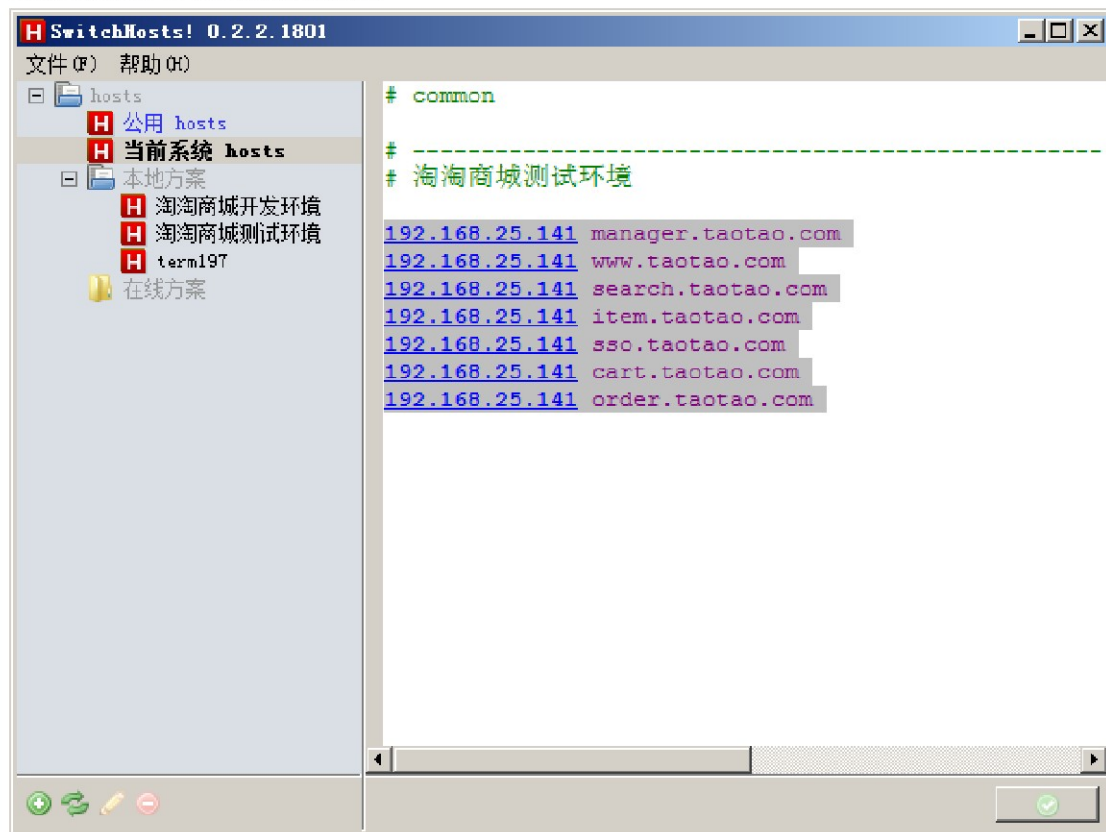
测试时使用域名访问网站，需要修改 host 文件。

所有的域名应该指向反向代理服务器。

配置 hosts 文件：



192.168.25.141 manager.taotao.com  
192.168.25.141 www.taotao.com  
192.168.25.141 search.taotao.com  
192.168.25.141 item.taotao.com  
192.168.25.141 sso.taotao.com  
192.168.25.141 cart.taotao.com  
192.168.25.141 order.taotao.com



反向代理的配置:

```
#user nobody;  
worker_processes 1;  
  
#error_log logs/error.log;  
#error_log logs/error.log notice;  
#error_log logs/error.log info;  
  
#pid logs/nginx.pid;  
  
events {  
    worker_connections 1024;  
}
```



```
http {
    include      mime.types;
    default_type  application/octet-stream;

    #log_format  main  '$remote_addr - $remote_user [$time_local] "$request" '
    #              '$status $body_bytes_sent "$http_referer" '
    #              '"$http_user_agent" "$http_x_forwarded_for"';

    #access_log  logs/access.log  main;

    sendfile      on;
    #tcp_nopush    on;

    #keepalive_timeout  0;
    keepalive_timeout  65;

    #gzip  on;

    upstream manager.taotao.com {
        server 192.168.25.137:8080;
    }
    upstream www.taotao.com {
        server 192.168.25.137:8081;
    }
    upstream search.taotao.com {
        server 192.168.25.137:8082;
    }
    upstream item.taotao.com {
        server 192.168.25.138:8080;
    }
    upstream sso.taotao.com {
        server 192.168.25.138:8081;
    }
    upstream cart.taotao.com {
        server 192.168.25.139:8080;
    }
    upstream order.taotao.com {
        server 192.168.25.139:8081;
    }

    server {
        listen      80;
        server_name  manager.taotao.com;
```



```
#charset koi8-r;

#access_log logs/host.access.log main;

location / {
    proxy_pass http://manager.taotao.com;
    index index.html index.htm;
}
}
server {
    listen 80;
    server_name www.taotao.com;

    #charset koi8-r;

    #access_log logs/host.access.log main;

    location / {
        proxy_pass http://www.taotao.com;
        index index.html index.htm;
    }
}
server {
    listen 80;
    server_name search.taotao.com;

    #charset koi8-r;

    #access_log logs/host.access.log main;

    location / {
        proxy_pass http://search.taotao.com;
        index index.html index.htm;
    }
}
server {
    listen 80;
    server_name item.taotao.com;

    #charset koi8-r;

    #access_log logs/host.access.log main;
```



```
        location / {
            proxy_pass    http://item.taotao.com;
            index    index.html index.htm;
        }
    }
    server {
        listen            80;
        server_name    sso.taotao.com;

        #charset koi8-r;

        #access_log    logs/host.access.log    main;

        location / {
            proxy_pass    http://sso.taotao.com;
            index    index.html index.htm;
        }
    }
    server {
        listen            80;
        server_name    cart.taotao.com;

        #charset koi8-r;

        #access_log    logs/host.access.log    main;

        location / {
            proxy_pass    http://cart.taotao.com;
            index    index.html index.htm;
        }
    }
    server {
        listen            80;
        server_name    order.taotao.com;

        #charset koi8-r;

        #access_log    logs/host.access.log    main;

        location / {
            proxy_pass    http://order.taotao.com;
            index    index.html index.htm;
        }
    }
}
```



```
}
```

### 3. 项目总结

## 4. 面试中的问题回答

### 4.1. 淘淘网站并发数：

1000-2000 左右并发。

### 4.2. 人员配置

产品经理：3 人，确定需求以及给出产品原型图。

项目经理：1 人，项目管理。

前端团队：5 人，根据产品经理给出的原型制作静态页面。

后端团队：20 人，实现产品功能。

测试团队：5 人，测试所有的功能。

运维团队：3 人，项目的发布以及维护。

### 4.3. 开发周期

采用迭代开发的方式进行，一般一次迭代的周期为一个月左右。

### 4.4. Sku

最小库存量单位。

Sku==商品 id

### 4.5. 电商活动倒计时方案：

1、确定一个基准时间。可以使用一个 sql 语句从数据库中取出一个当前时间。SELECT NOW();



- 2、活动开始的时间是固定的。
- 3、使用活动开始时间-基准时间可以计算出一个秒为单位的数值。
- 4、在 redis 中设置一个 key（活动开始标识）。设置 key 的过期时间为第三步计算出来的时间。
- 5、展示页面的时候取出 key 的有效时间。Ttl 命令。使用 js 倒计时。
- 6、一旦活动开始的 key 失效，说明活动开始。
- 7、需要在活动的逻辑中，先判断活动是否开始。

## 4.6.秒杀方案：

- 1、把商品的数量放到 redis 中。
- 2、秒杀时使用 decr 命令对商品数量减一。如果不是负数说明抢到。
- 3、一旦返回数值变为 0 说明商品已售完。