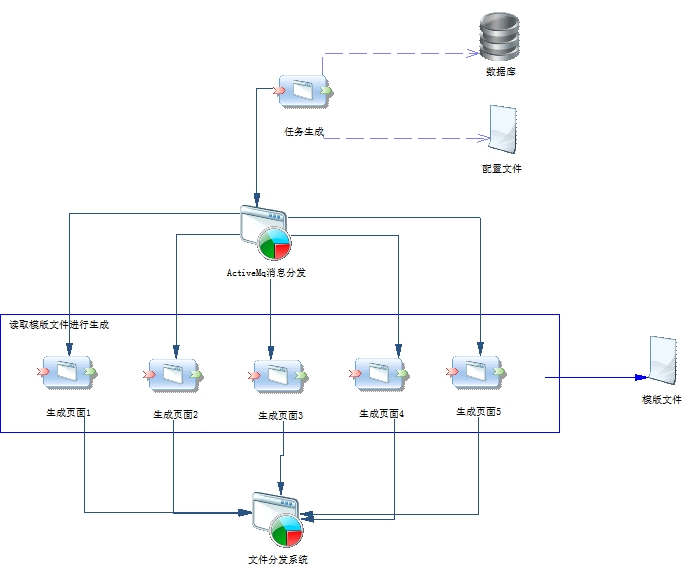
## 一、基本架构



第一步：

通过一个定时任务或手工触发的形式，从数据库和配置文件读取数据生成静态化任务；

第二步：

通过activemq进行任务分发，通过p2p消息发布到各个生成服务上；

第三步：

各个生成节点读取freemarker模版文件生成html页面；

第四步：

通过文件分发系统，将生成的html页面发布到各个web容器上。

## 二、触发条件及生成规则

1. 要分类别进行静态化。

比如商品部分，比如帮助部分，需要有不同的模版文件。

数据可以按分类进行静态化，也可以全部静态化。参考shopxx的实现。

1. 静态化数据控制。

A．先判断模板是否发生了变化。

模板要和静态化类别关联，如果模板发生了变化，则全部该类别下的数据都要进行静态化。

B. 每条数据的静态化标志，用于标志该数据是否需要进行静态化。

C. 可以选择数据记录进行静态化。

D． 新增后立即进行静态化，更新后立即进行静态化。

这样A,B,C三项处理就仅仅用来处理静态化不成功的记录。

1. 数据库变化。

增加字段：静态化后的URL，是否需要静态化。

新增数据表：模板记录表，模板id，模板路径，模板是否发生了变化。

1. 设定定时任务执行。

除了手工操作以外，还允许每晚01：00自动执行。

1. 静态文件的生成策略：首页和频道页、概览页定时生成（这些页面延时几分钟是没问题的），细览页面实时生成（避免生成概览静态时细缆还没生成造成问题）。对于细缆页面在后台发布信息的时候实时调用生成静态页面的接口，产生静态文件，并将细缆页面的静态URL存入到相应的数据表的相应记录中，修改的时候也重新生成修改，是否允许删除呢？

某些商品如果被删除，则要重新生成首页，频道页面，并删除生成的该商品的细览页面。

1. 静态文件名称目录设计原则：对于频道页和概览页我们采用固定目录和固定静态名，每次定时生成静态页面的时候就覆盖，（用户点击那一刻后台覆盖静态文件是否会出现问题需要验证一下，网上说没问题）。这样做的好处是避开了频道和概览静态文件的管理，简化开发。
2. 定时任务生成最热推荐，品牌推荐，同类热卖，频道页面。

## 三、其他

静态页面可以放入CDN缓存。

站点的问题需要考虑进去，比如北京站，杭州站风格应该会不同。



Web服务器采用nginx,处理静态文件效率会高一些。

静态页面 采用ssi技术进行包含,支持shtml后缀和html后缀。

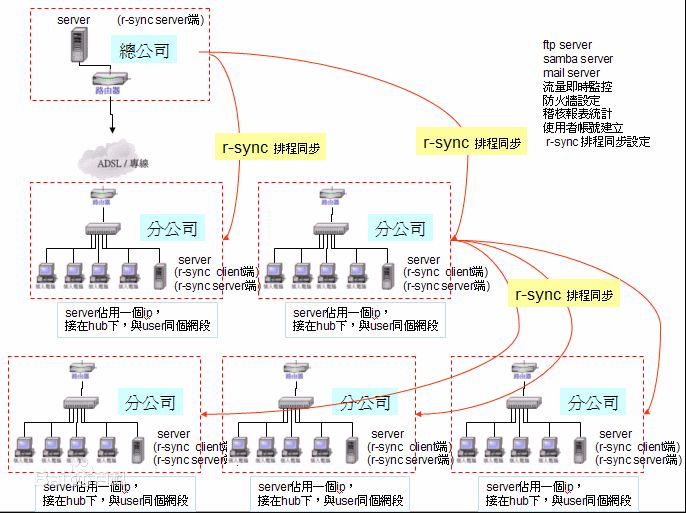
## 四、绘制关键时序图



## 五、模版文件,html文件的同步问题

Rsync、goodsync等。决定采用rsync,可以增量同步。





## 六、 主要模块划分



## 七、CMS部分

参考jeecms的实现。

## 八、数据库设计





## 九、其他问题

1. 分站问题

访问路径分开，绑定二级域名。

每个分站一套模版，。

标准CMS系统。

2. 伪静态。

主要是搜索部分和频道部分用到。

3. SEO优化。

主要存在标题需要变化，域名合并（或通过apache转发来实现），频道的深度要限制在1-2层，不能太深。

4. 促销用的自定义标签。

给商品打tag，可以用于搜索商品。

5. 在页面上调整模版。

6. 监控代码的处理。

7. 购物车不能静态化，静态和动态的分开处理。

## 十、工作计划

5月26日上线。