1.大赛主页

[**http://cloud.seu.edu.cn/contest/index**](http://cloud.seu.edu.cn/contest/index)

作品提交日期是12月15日

分为3种比赛形式

我们选择的是 命题赛

我们选择的题目是第1题 见文件MT1.PDF

2.我们队伍报名信息：

**团队名称 UNO**

**指导教师 李江峰**

**张静宵 15121033474 honestboye@aliyun.com**

**陈复新 18818203198** [**5217wyx@gmail.com**](mailto:5217wyx@gmail.com)

**周游 13761085703** [**zyrusher@gmail.com**](mailto:zyrusher@gmail.com)

**3.命题赛评分标准**

**命题赛**

命题赛主要考察命题需求的完成程度，主要从需求的完成程度，技术路线，文档规范以及创新性几个方面考量。具体标准如下：

**需求完成程度（40%）**：必选需求需全部完成，否则不得分，可选需求根据完成程度进行加分。

**方法和技术路线（30%）**：技术路线合理，方法符合命题要求。

**文档规范(10%)**：文档格式规范，思路清晰，表达清楚，能够充分体现程序设计思路，实现方案，测试结果。

**成果展示(10%)**：演示内容丰富，展现方式多元化。

**创新性(10%)**：对命题中的需求有进一步深层次分析，且分析合理，问题明确，提出的解决方法合理有效，有创新性。

命题赛和创意赛都要提交你的相关设计方案的WORD文档、说明设计思路的PPT，以及可运行的demo、原型系统或算法。

**4.命题赛评分标准**

现阶段思路

1. 讨论了每个必选需求的方案。
   1. 存储平台：HDFS上的存储方案，HBase
   2. 管理功能：HTML+PHP+REST

REST：

* 1. 图片压缩：压缩算法？图片编码？

Java-封装成接口

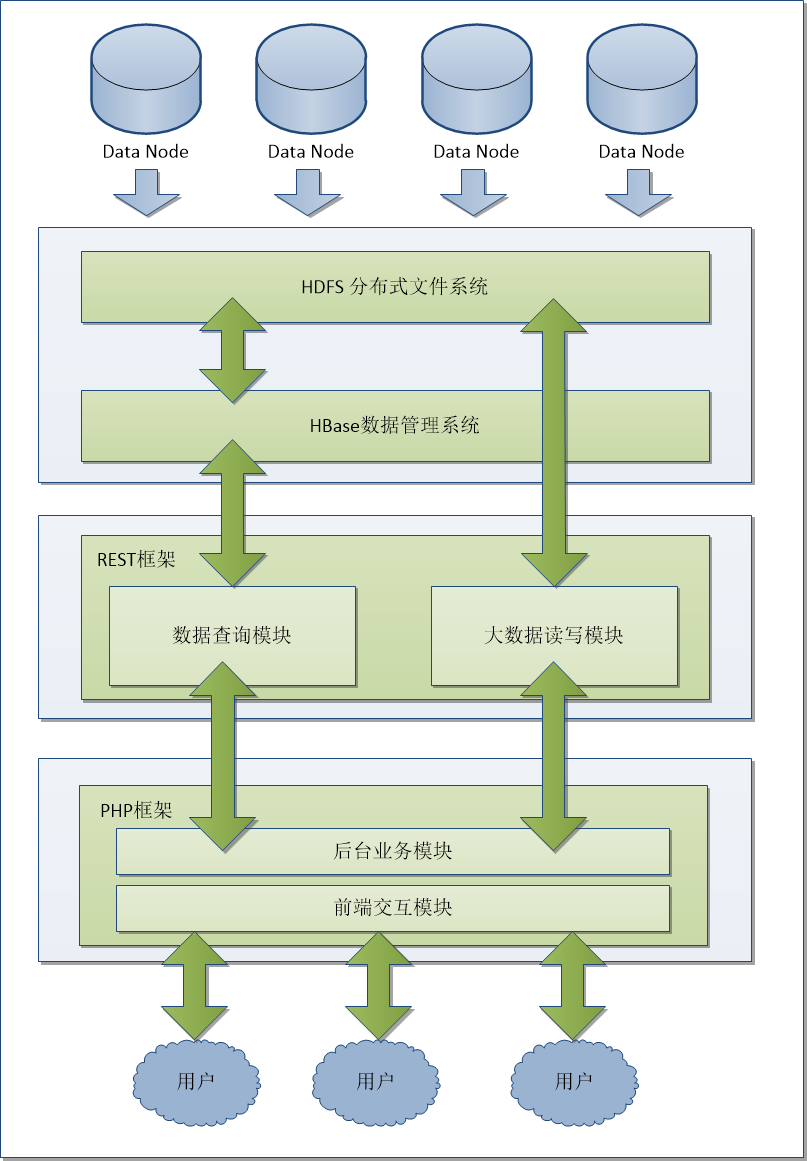
* 1. 在线裁剪：HTML5前端实现裁剪，如需保存再传到后台

前端

* 1. 在线水印：前端可实现添加水印，如需保存再传到后台

1. 可选需求：
   1. 高清展示：高清图片如何传输和存储？
   2. 全景
2. 需要找出1~2个亮点，以及设计一个场景，让这个系统体现出价值。可以从访问图片的效率、采用webservice提供可扩展性等方面来考虑。
   1. HDFS提供了对大数据存储的平台，但在HDFS上对数据的查询并不高效。一种建议：预测近期会访问到的数据，进行预取提高效率。这个网站是 一个电商网站，可以通过点击、交易的数据进行分析，将那些有较高概率在将来会被点到的图片预测为关键数据，进行预取。
3. 向与云计算、大数据相关的问题靠拢

11月28日讨论



11月28日的讨论，主要确定了基本架构，明确了下一阶段的分工。

下次讨论时间12月5日

# 架构图如上图所示，下面分层介绍：

1. 数据层：
   1. 物理节点：物理上有多个分布式数据节点来支持整个存储系统
   2. HDFS文件系统：用HDFS来进行图片数据的存储和读写操作
   3. HBase数据管理系统：采用HBase（或再加上Hive）来存储用户数据以及实现数据增删改查的等操作（但图片数据由于其体积大，直接访问HDFS操作）。
2. 中间层：

采用REST框架封装服务，分为两个模块

* 1. HBase数据查询模块：提供操作Hbase的服务，能够获取用户信息，图片地址
  2. HDFS文件操作模块：提供HDFS读写操作，用于图片数据的存取操作

1. 业务层：

采用PHP框架开发一个网站，向用户提供管理接口，同时利用中间层提供的服务对数据层进行操作

* 1. 后台业务模块：相应用户请求，处理业务逻辑
  2. 前端交互模块：为用户提供交互界面，向后台业务模块传递请求

# 功能需求及分工：

## 必选功能需求列表

|  |  |  |  |
| --- | --- | --- | --- |
| No | 功能 | 架构层 | 工作安排 |
| 1 | 数据存储 | 数据层 | 张：HDFS/HBASE存储环境搭建。Hbase数据模型设计。 |
| 2 | Hbase操作模块 | 中间层 | 张：Hbase操作的Java接口 |
| 3 | HDFS文件操作模块 | 中间层 | 周：HDFS文件操作Java接口 |
| 4 | REST服务封装 | 中间层 | 周：将2、3两项封装成REST服务 |
| 5 | 压缩后台 | 业务层 | 陈：PHP库实现；周：压缩算法调研 |
| 6 | 裁剪、水印后台 | 业务层 | 陈：PHP库实现 |
| 7 | 与REST服务集成 | 业务层 | 陈、周：在PHP后台调用REST服务实现对数据层的访问 |
| 8 | 压缩前端  裁剪、水印前端 | 业务层 | 陈：HTML5的前端设计和实现 |
| 9 | 用户管理操作界面 | 业务层 | 陈：利用网页模板等实现包括上传、展示、删除、压缩、裁剪、水印等界面 |

## 文档工作

张：总体架构、数据层

周：中间层

陈：代码工作量比较大，暂略

## 可选功能及创新点：

1. 高清图片：
   1. 读取的时候拆成几块，像google map那种分块传输到客户端
   2. 压缩成多个尺寸，一开始读取小尺寸，用户点了放大，再把放大的读出来
2. 全景：资源、技术所限，暂不考虑
3. 缓存：设立缓存服务器上的缓存。对商户的图片设置标签

设定一些列的规则，满足规则时权重为1，否则为0，算总分，将总分最高的图片认为是最有可能被访问的，存入缓存中。

1. 用户层的预取：通过用户习惯进行图片预取

## 问题场景：

1.高清图片：在网络连接速度较低的 移动设备上，需要放大的时候才读，放大的时候会进行 裁剪，只显示它需要的那一块。

2．缓存：访问量大的时候，或者服务器响应时间很长，如果有本地的缓存服务器，会更快。

难点：训练图片资源的评分模型（数据集？）？缓存服务器在技术上如何搭建？缓存服务器如何分布？

3. 用户层预取：和上面一样。某用户现在浏览商品A，系统预测下一次会看商品B，这时候将B相关的图片预取到浏览器，当用户真的点了B的时候，速度非常快，利用用户浏览A的时间，将B进行了预取，对用户而言就像B的加载非常快一样。难点：如何预测商品B？简单的推荐模型？

总体目标/亮点：

亮点是指采用了我们的创新点能够做到与别人不同的地方。通过缓存和预取技术，整个系统的响应速度更快，用户体验提升。如果能够成功，实际上已经包含了一个预测模型来判断下一个将要读取的数据。

# Schedule：

|  |  |
| --- | --- |
| Event | Date |
| 完成必须需求 | 12月5号 |
| 完成创新点 | 12月12号 |
| 完成文档和需要上交的材料 | 12月15号 |

# 开发环境：

1. 系统：linux Ubuntu
2. Hadoop 0.2.0
3. JDK 1.6