图片优化方案

目前上传的图片由于都是扫描件，原图尺寸非常大，而且在java服务器处理图片时占用的内存与CPU也非常大。

仔细研究浏览器访问图片的过程。

1 浏览器打开与JAVA服务器的连接，发送请求给JAVA服务器。

2 JAVA服务器根据请求参数中的图片ID去查找数据库

3 根据数据库中的图片路径去查找磁盘，读取图片文件

4 将文件内容加入内存，在内存中进行图片格式转换

5 将格式转换后的图片返回给浏览器

6 关闭与浏览器的网络连接，整个过程结束。

在这6个步骤中，第一个步骤不建议优化。

因为这个网络协议使用的是HTTP协议，虽然有更好的web socket协议，但是web socket协议是HTML5标准，只有少数浏览器支持，而且市场技术也不太成熟。

目前能优化的是使用HTTPS协议，可以避免图片被劫持，避免部分浏览器（比如IE）的兼容性问题。

第二个步骤，查找数据库的单条数据是不需要优化的，因为数据库技术从1970年代发展至今，这点还是可以保证的。

3 读取文件。

这一步是我们要优化的重点，读取文件的内容要尽量小。所以需要把上传的图片进行处理，但是过分的处理会导致上传过程变得缓慢。

所以这个过程一定是异步的，在上传文件时，先快速保存一份原始文件。然后来开启一个异步任务。

在这个异步任务中，做3件事情。

第一件事情，

将图片按正确的图片格式进行编码。

第二件事情，将大图片按比例缩小。

统一为高度400像素。

第三件事情，计算图片的MD5摘呀与SHA1摘要，过滤重复的图片

第4、5步 渲染图片

由于图片已经格式化，所以优化的这一步就只需要直接将图片从文件加载到内存直接发送到网络流就可以了。

6 关闭连接。具体不同的通讯协议（HTTP\HTTPS\WEB SOCKET）有所不同。