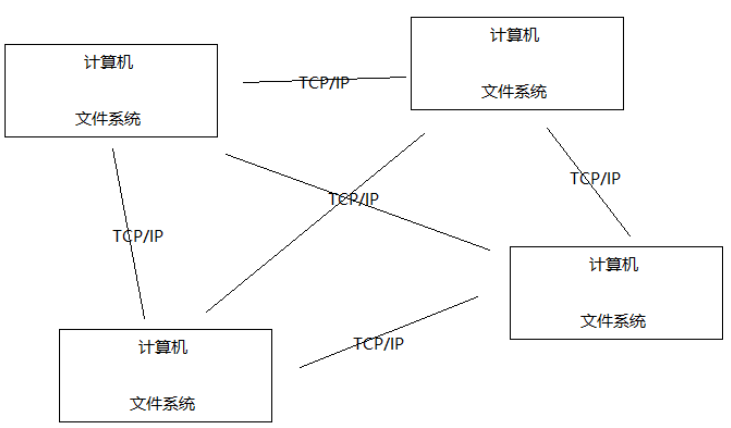


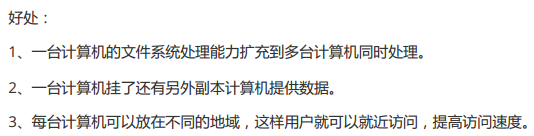
文件系统是操作系统的一部分

**分布式文件系统**

分布式文件系统是指文件系统管理的物理存储资源不一定直接连接在本地节点上 ，而是通过计算机网络与节点相连。





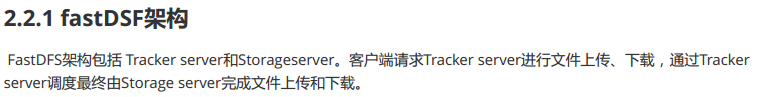


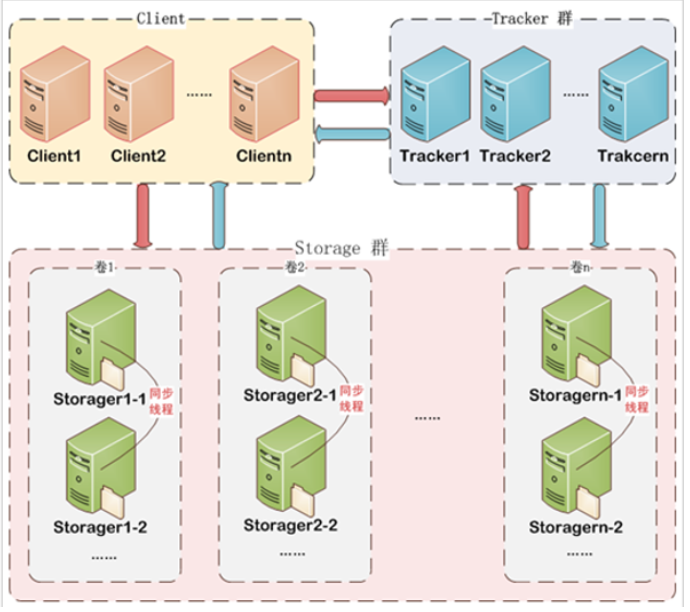
常见分布式文件系统：GFS、HDFS

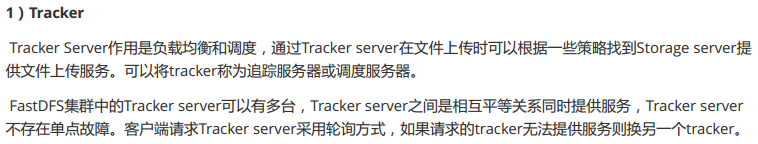
特点：都采用**主从结构**，由一个存储文件元数据和分块信息的主节点和多个存储文件数据的从节点构成。一个完整的文件将会被分成若干块存储在多个数据节点(从节点)中。客户端先从主节点获取文件的元信息和数据分块信息，然后就可以去对应的从节点（数据节点）存取数据。(有一个分块和合并的过程)

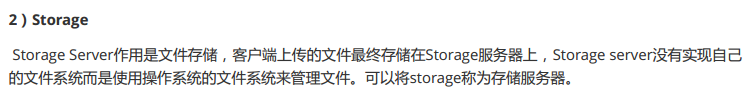


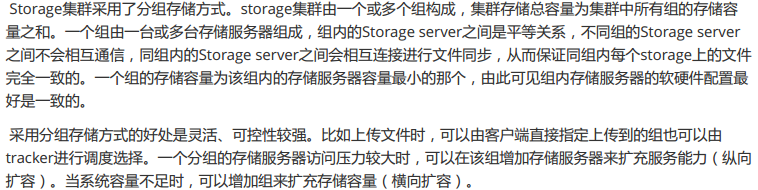
fastDFS特点：系统复杂性低，性能高，适合存储图片等这种小文件，不对文件进行分块，没有分块与合并的开销，使用socket进行通信，通信速度很快。

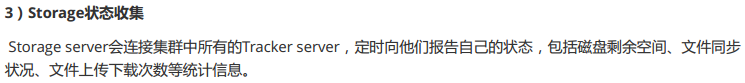


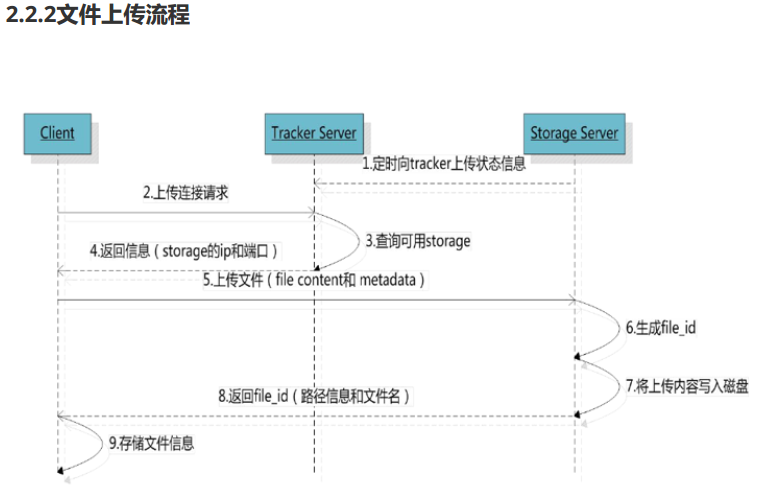






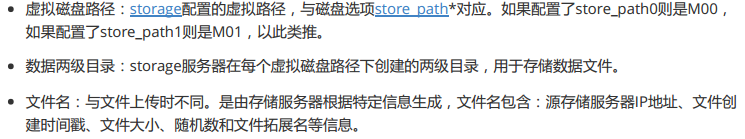


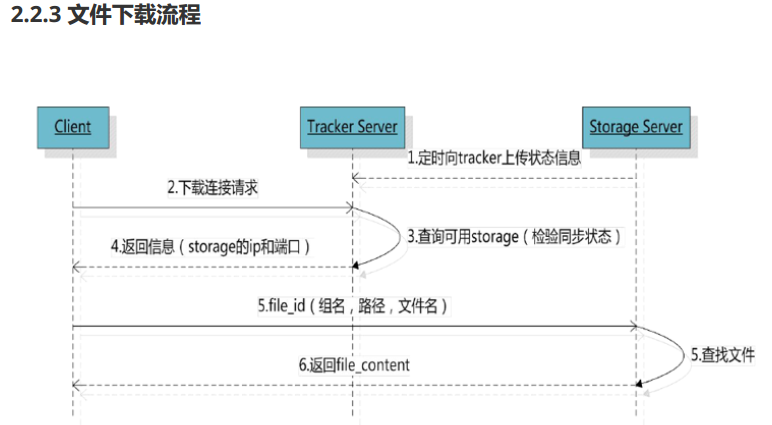


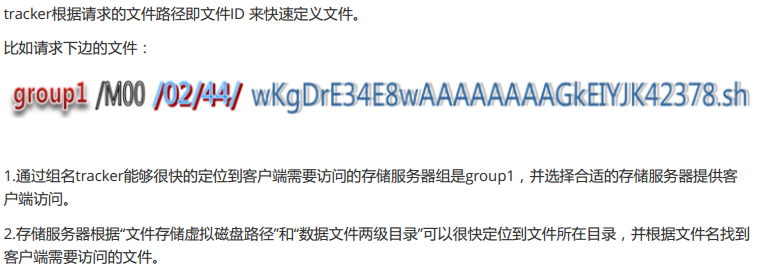




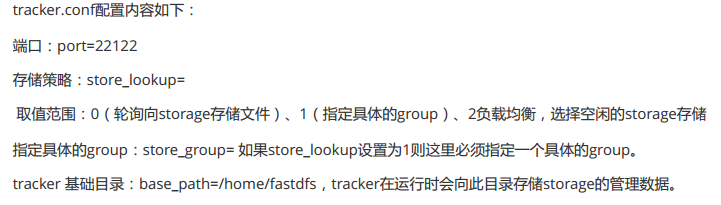


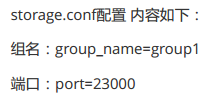






配置文件





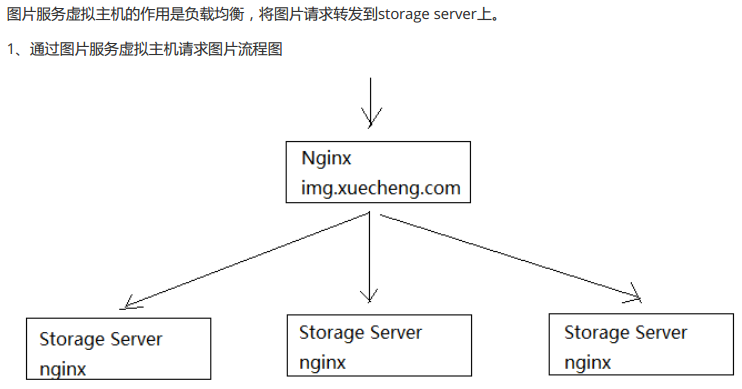




FastDFS通过Tracker服务器,将文件放在Storage服务器存储,但是同组之间的服务器需要复制文件,有延迟的问题.假设Tracker服务器将文件上传到了192.168.1.80,文件ID已经返回客户端,这时,后台会将这个文件复制到192.168.1.30,如果复制没有完成,客户端就用这个ID在192.168.1.30取文件,肯定会出现错误。这个fastdfs-nginx-module可以重定向连接到源服务器取文件,避免客户端由于复制延迟的问题,出现错误。

当本机storage没有要找的文件时，转发到本组内其他storage server去寻找。





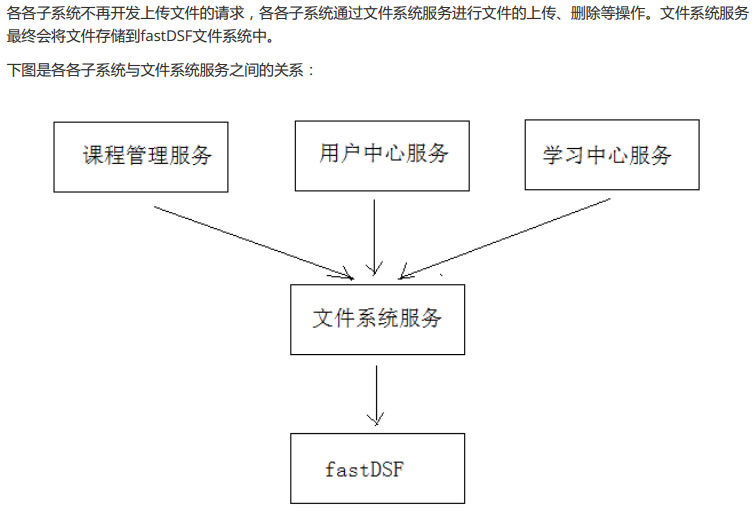


这样可以使得通过域名访问图片，如果有多个组，就配置多个group。

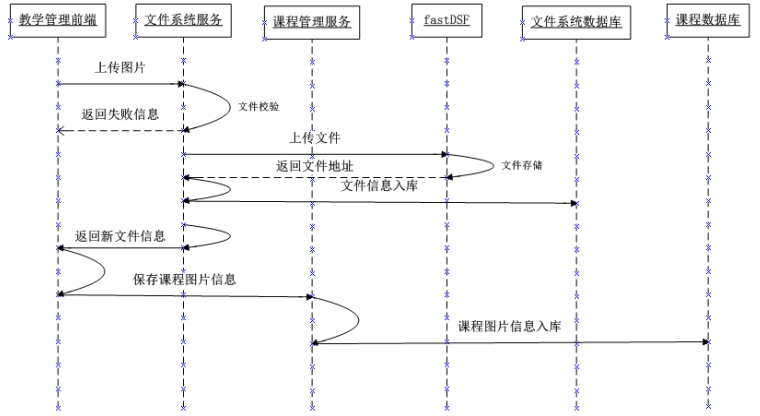


**上传图片开发需求分析**

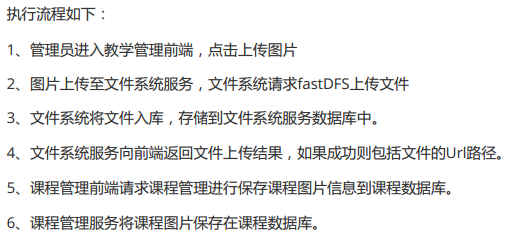
将文件上传功能单独抽取出来形成一个服务，其他服务来调用这个服务来完成文件上传。



上传图片处理流程：



文件系统服务有自己的数据库(MongoDB)用于存储由它上传的文件信息，相当于备份。它也会返回该文件的信息给其他系统，其他系统要也要去保存属于自己那部分的文件信息。





还有查询图片和删除图片功能，接口的参数都是courseId