

视频流系统设计

Video Streaming System

主讲人 南帝老师



Scenario场景

设计一个 YouTube

1. 了解 YouTube 包含哪些功能
2. 这些功能背后的挑战

YouTube 是一个全球最大的视频分享网站之一

Functional Requirements

用户可以上传视频
用户可以观看和分享视频
生成^{thumbnail}缩略图
用户可以点赞、评论

Non-Functional Requirements

系统的高可用
观看视频的流畅度
视频的实时推荐

月活跃用户(MAU)



日活跃用户(DAU)



每天观看视频数量



每分钟上传视频时长



用户平均观看时间



如果一个人24小时不停歇地看视频，则看完 Youtube 一天内上传的视频需要82年。

— 截止2019年5月

数据来源：

- https://en.wikipedia.org/wiki/List_of_content_platforms_by_monthly_active_users
- <https://www.androidauthority.com/youtube-stats-1016070/>

- 根据数据，我们有150M的日活跃用户(DAU)，如果用户平均每天观看30个视频，则每秒视频的观看数为：

$$150M * 30 / 86400 \text{ 秒} = 52083 \text{ 视频/秒, 即每天视频观看数为45亿}$$

- 根据经验值上传视频数总是小于观看视频数，因此假设每上传一个视频，我们就会观看500部视频，则每秒视频的上传数为：

$$52083 / 500 = 104 \text{ 视频/秒}$$

- 假设平均每个视频时长为5分钟，则每秒上传的视频时长为：

$$104 * 5 = 520 \text{ 分钟/秒, 即每分钟上传视频时长520小时}$$

- 存储空间估算：假设在平均的情况下，一分钟视频需要50MB的存储空间（视频需要以多种格式存储），则一秒钟内上传的视频所需的总存储量为：

$$520 * 50 = 26000 \text{ MB/s} = 26 \text{ GB/s}$$

- 带宽估算：每秒上传520分钟的视频，并假设每次上传的视频占用的带宽为 166KB/s，那么：

$$520 * 60 * 166 = 5179200 \text{ KB/s} = 5 \text{ G/s}$$

Service服务

了解 YouTube 相关服务

用户服务


UserService

查询用户基本信息

上传服务

UploadService

视频上传

 转码服务

EncodeService

视频转码

缩略图服务

ThumbService

缩略图生成

视频服务

VideoService

视频播放
缩略图加载

Storage存储

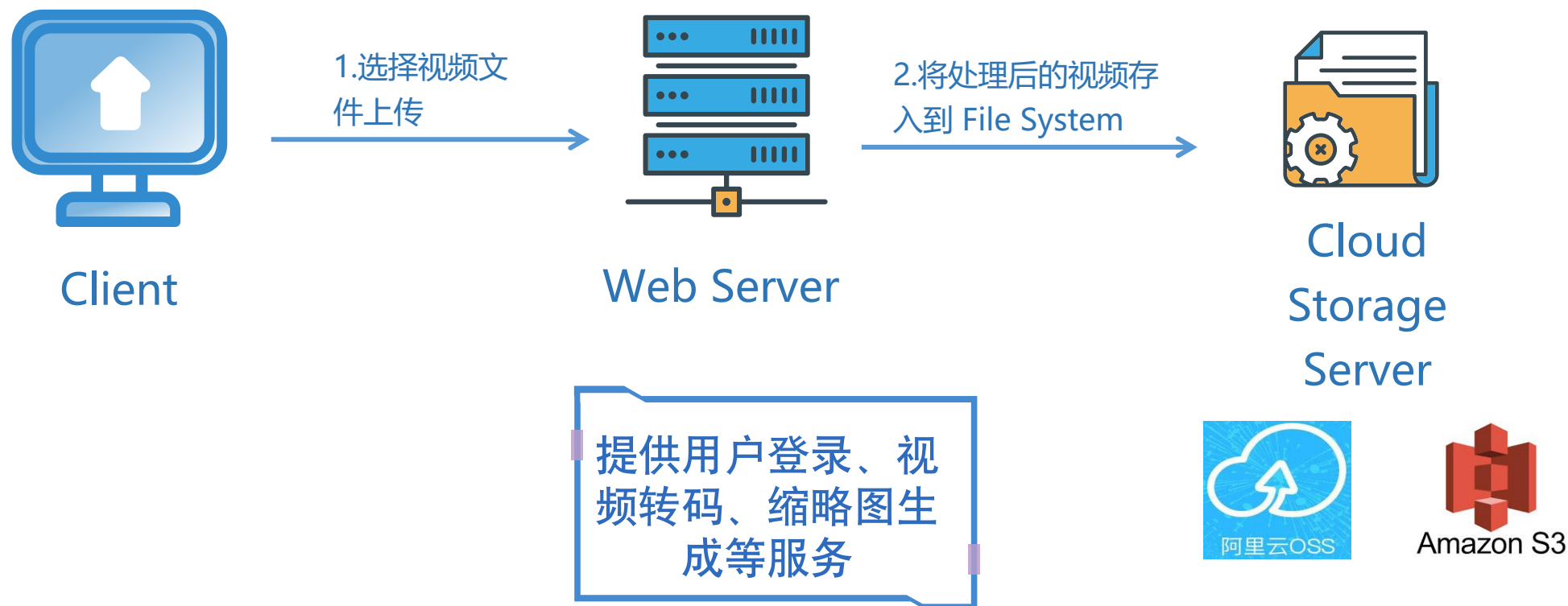
模型的设计

YouTube 模型定义

01

视频上传

一个简单的视频上传流程



将文件上传到 Web Server 的过程中会有什么问题？

在同一个请求中，要上传大量的数据，导致整个过程会比较漫长，且失败后需要重头开始上传。

解决方案：

Sharding
视频切分

断点续传

Sharding 如何对视频进行切分?

- 在编码方式上传中，在前端我们只要先获取文件的二进制内容，然后对其内容进行拆分，最后将每个切片上传到服务端即可。
- 在JavaScript中，文件File对象是Blob对象的子类，Blob对象包含一个重要的方法slice，通过这个方法，我们就可以对二进制文件进行拆分。

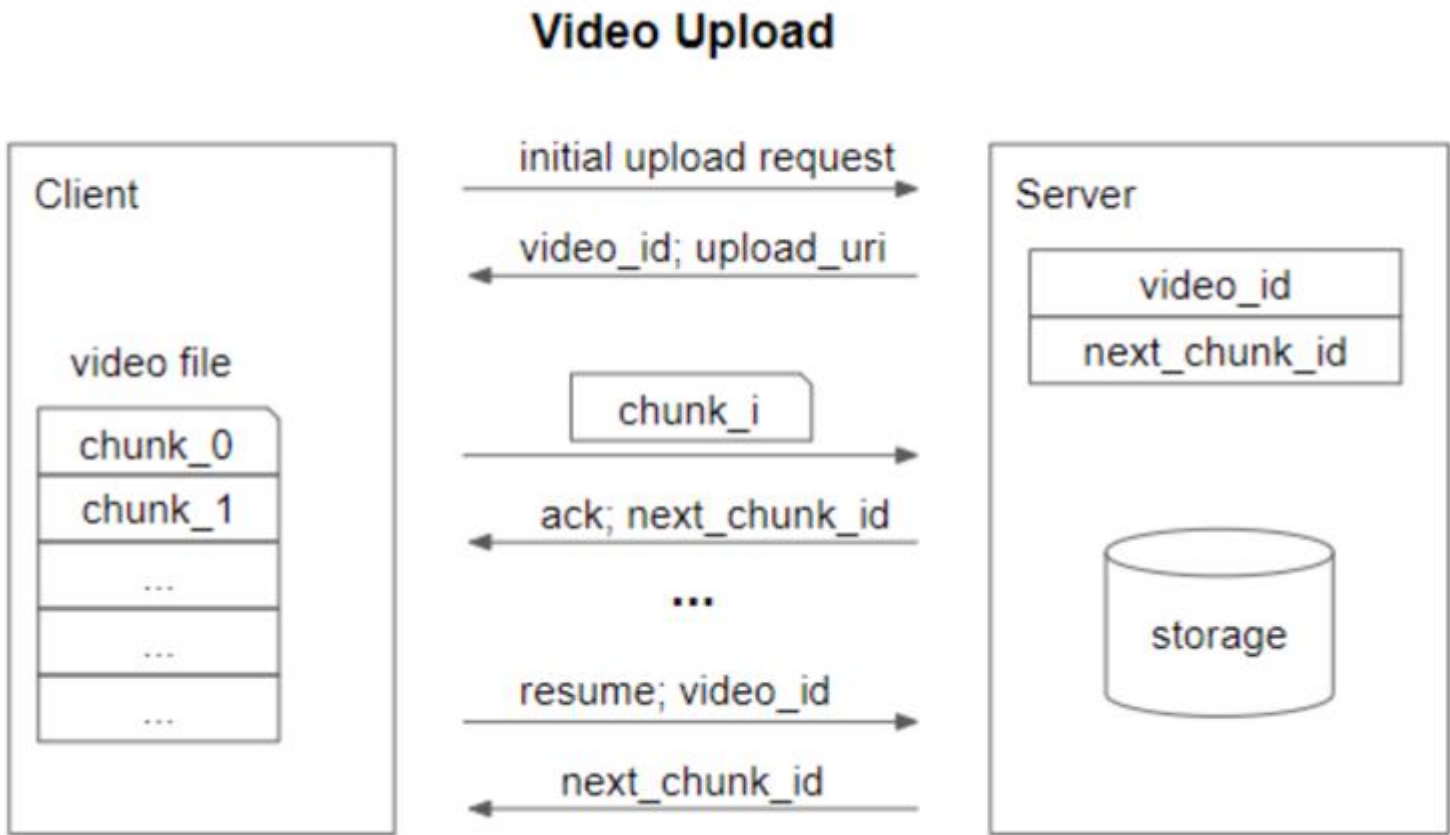
```
function slice(file, piece = 1024 * 1024 * 5) {  
  let totalSize = file.size; // 文件总大小  
  let start = 0; // 每次上传的开始字节  
  let end = start + piece; // 每次上传的结尾字节  
  let chunks = []  
  while (start < totalSize) {  
    // 根据长度截取每次需要上传的数据  
    // File对象继承自Blob对象，因此包含slice方法  
    let blob = file.slice(start, end);  
    chunks.push(blob)  
  
    start = end;  
    end = start + piece;  
  }  
  return chunks  
}
```

Upload Request(HTTP 1.1)


- 视频切分后，在客户端根据哈希算法生成所有视频切片的 chunk_id
- 客户端发起上传请求，将所有 chunk_id 带到服务端
- 服务端生成 video_id & 保存目录并返回给客户端
- 客户端开始上传 chunk

PS:

- 当上传完一个 chunk 之后，服务端告诉客户端需要传下一个 chunk



视频上传的过程中，如果用户主动停止上传，那么已经上传的部分如何处理？

1. 一直存在服务器上，等待用户下次接着上传
2. 会存一段时间，过期后删除上传部分视频 
3. 直接删除

记录视频上传最后的时间，超过一定时间范围将上传视频清空

02

Transcoding

视频转码和

Thumbnail

缩略图生成

为什么对视频进行转码？
Transcoding

格式转码

用户可能上传各种格式的视频，需要转换成 Youtube 兼容的格式在网页端进行播放。

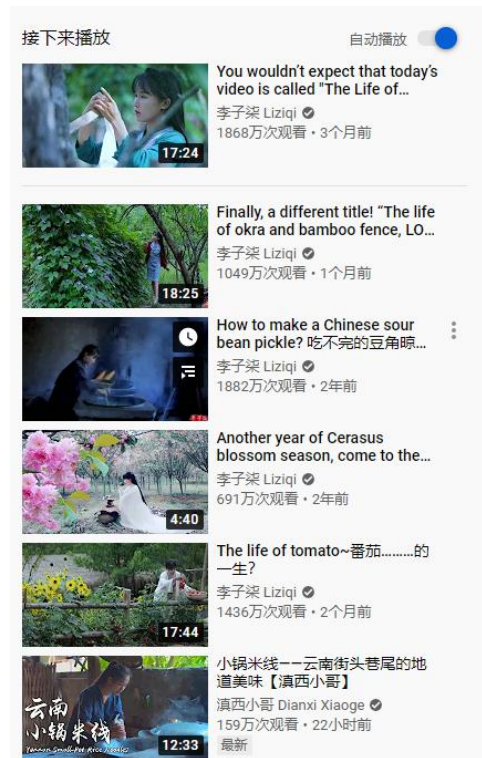
清晰度转码

用户根据网络环境的不同，会有不同清晰度视频的需求。



进度条缩略图

可以让用户根据缩略图判断
要定位的进度条位置

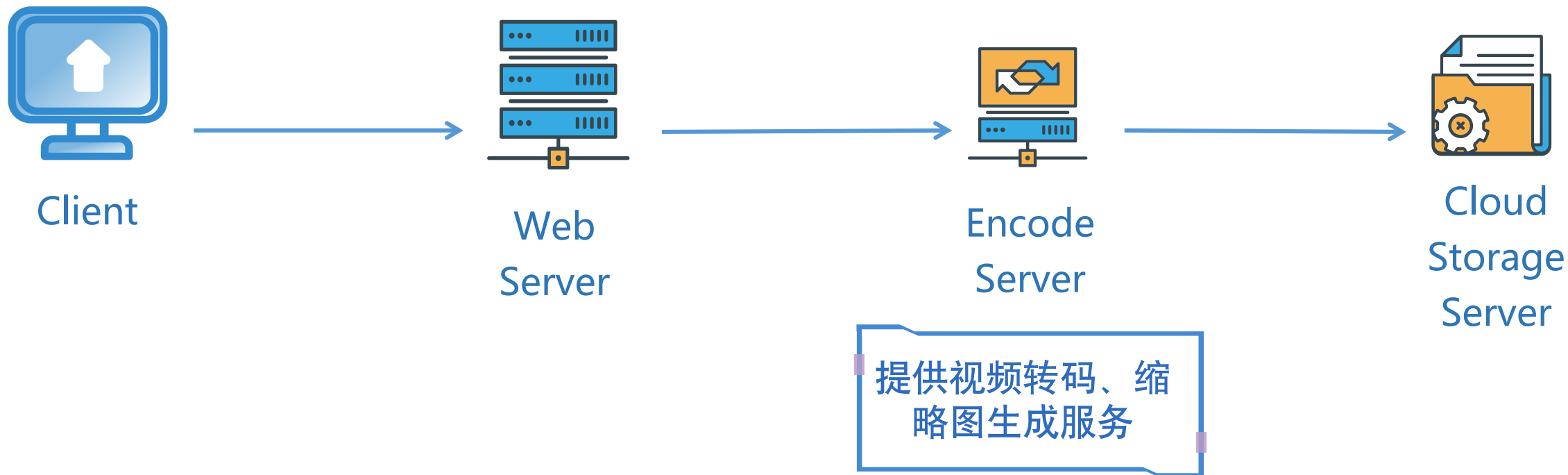


封面缩略图

预览视频的大致内容

视频转码和缩略图生成如何实现？

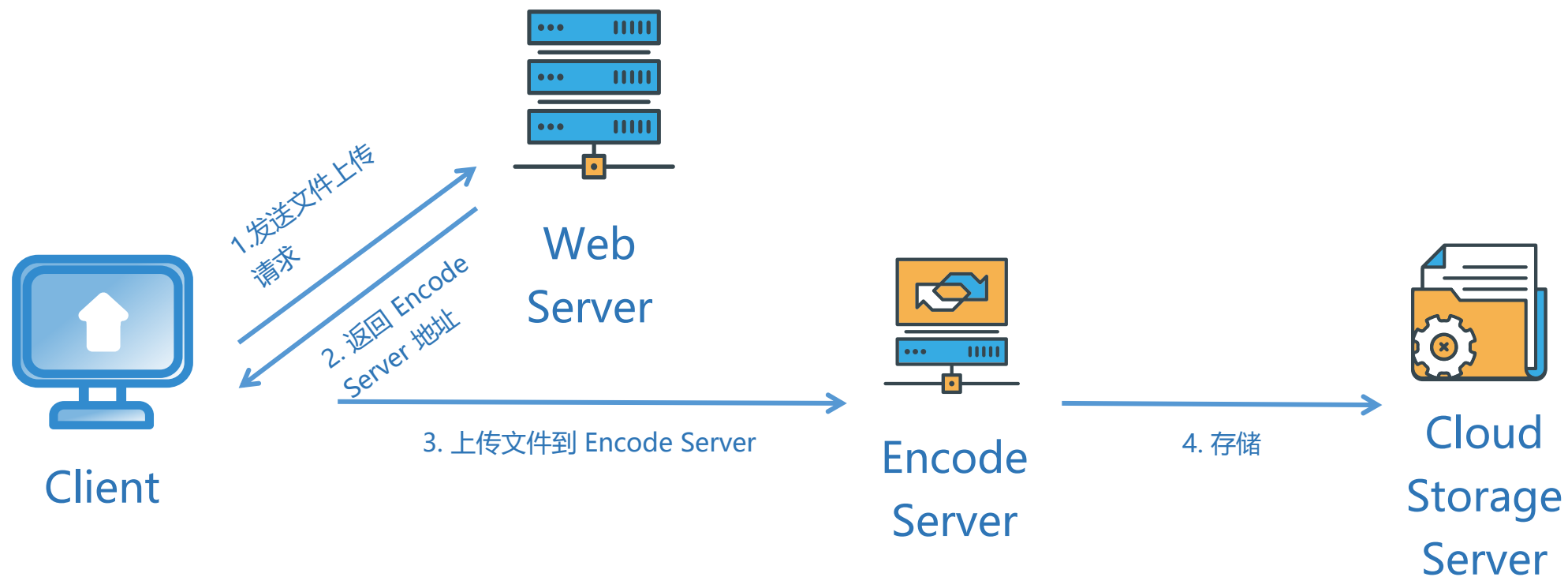
视频转码和缩略图生成实际上应该交给一个专门的服务器来实现，这个服务器就是 Encode Server。



视频转码和缩略图生成服务可以分开吗？

再思考一个问题，上面的流程还有哪里可以优化？

最好还是放在一台 Server 上，因为视频在系统内部的传输也要占用带宽。



这样做的目的同样也是为了节省带宽。

Thumbnail 03 视频与缩略 图的存储



You wouldn't expect that today's video is called "The Life of Potato"! 你们都没想...
李子柒 Liziqi
1873万次观看 · 3个月前



4 Hours of Ambient Study Music To Concentrate - Improve your Focus and...
Quiet Quest - Study Music
697万次观看 · 6个月前



Fast and Furious 9 Official Trailer HD/2020.Coming Soon.fan made.
fully loaded
3393万次观看 · 2年前



Goblin 도깨비 OST (Chanyeol, Punch) - Stay with me MV
Crazy Banana
45万次观看 · 3年前



[ENG SUB] 달지않고 촉촉한 도시락케이크 만들기 By Sunday baking 미니 레터...
Daram Table다람테이블
61万次观看 · 3周前



RAINY DAY COFFEE SHOP AMBIENCE: Piano Music and Rain Sounds With...
Autumn Cozy
70万次观看 · 1个月前



【二米炊烟】Pepper 花椒，一種最能代表中國川菜的調味料。麻婆豆腐/夫妻...
二米炊烟ErmiChuiyan
69万次观看 · 1周前



Friends: Best Moments From Season 2 (Mashup) | TBS
TBS
191万次观看 · 3周前

思考：如何显示视频的标题，作者，播放时长等信息呢？

视频表 Video Table

字段 column	类型 type	解释 explain	例子 example
video_id	Primary Key, VARCHAR	主键	'V0001'
hash_code	VARCHAR	哈希值	'a8LjzD94D0c9z69b'
resolution	VARCHAR	清晰度	'1080P'
size	INTEGER	视频大小, 单位KB	545259520
duration	INTEGER	时长, 单位秒	459
metadata	BLOB	元数据	{ language:xxx tags:xxx category:xxx }

为什么metadata要存json字符串字段，有什么好处？



视频表为什么需要 hash_code 字段？

为了视频去重，避免上传重复的视频文件。

视频分片表 Chunk Table

字段 column	类型 type	解释 explain	例子 example
chunk_id	Primary Key, VARCHAR	主键, 哈希值	'd90dJhV8KN7hg24W'
video_id	Foreign key, VARCHAR	外键	'V0001'
start_time	INTEGER	开始时间, 单位秒	10
end_time	INTEGER	结束时间, 单位秒	20
folder	VARCHAR	存储路径	'/0001/'
resolution	VARCHAR	清晰度	'1080P'

缩略图表 Thumbnail Table

字段 column	类型 type	解释 explain	例子 example
thunb_id	Primary Key, VARCHAR	主键	'T0001'
video_id	Foreign key, VARCHAR	外键	'V0001'
folder	VARCHAR	缩略图目录	'/mmmm/01'
size	INTEGER	缩略图大小, 单位KB	10
type	VARCHAR	缩略图类型	'progress bar'
moment	INTEGER	进度条时刻,单位秒	35

Video Table 和 Chunk Table, Thumbnail Table 的关系是什么?

1. 一对一

2. 一对多



3. 多对多

用户表 User Table

字段 column	类型 type	解释 explain	例子 example
user_id	Primary Key, VARCHAR	主键	'U0001'
user_name	VARCHAR	用户名	'no one'
gender	INTEGER	性别, 1表示男, 2表示女, 0表示未填	0

用户视频关系表 User Video Table

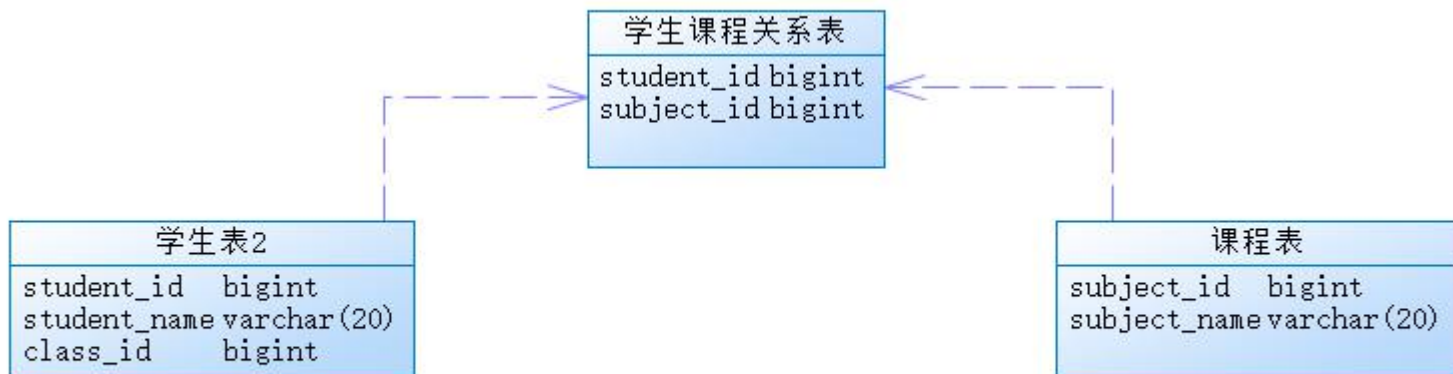
字段 column	类型 type	解释 explain	例子 example
user_id	Foreign Key, VARCHAR	外键	'U0001'
video_id	Foreign Key, VARCHAR	外键	'V0001'
create_time	DATETIME	视频上传时间	2020/09/0 09:00:00
metadata	BLOB	元数据	{ title:xxx }

Video Table 和 User Table 的关系是什么?

- 1. 一对一
- 2. 一对多
- 3. 多对多 

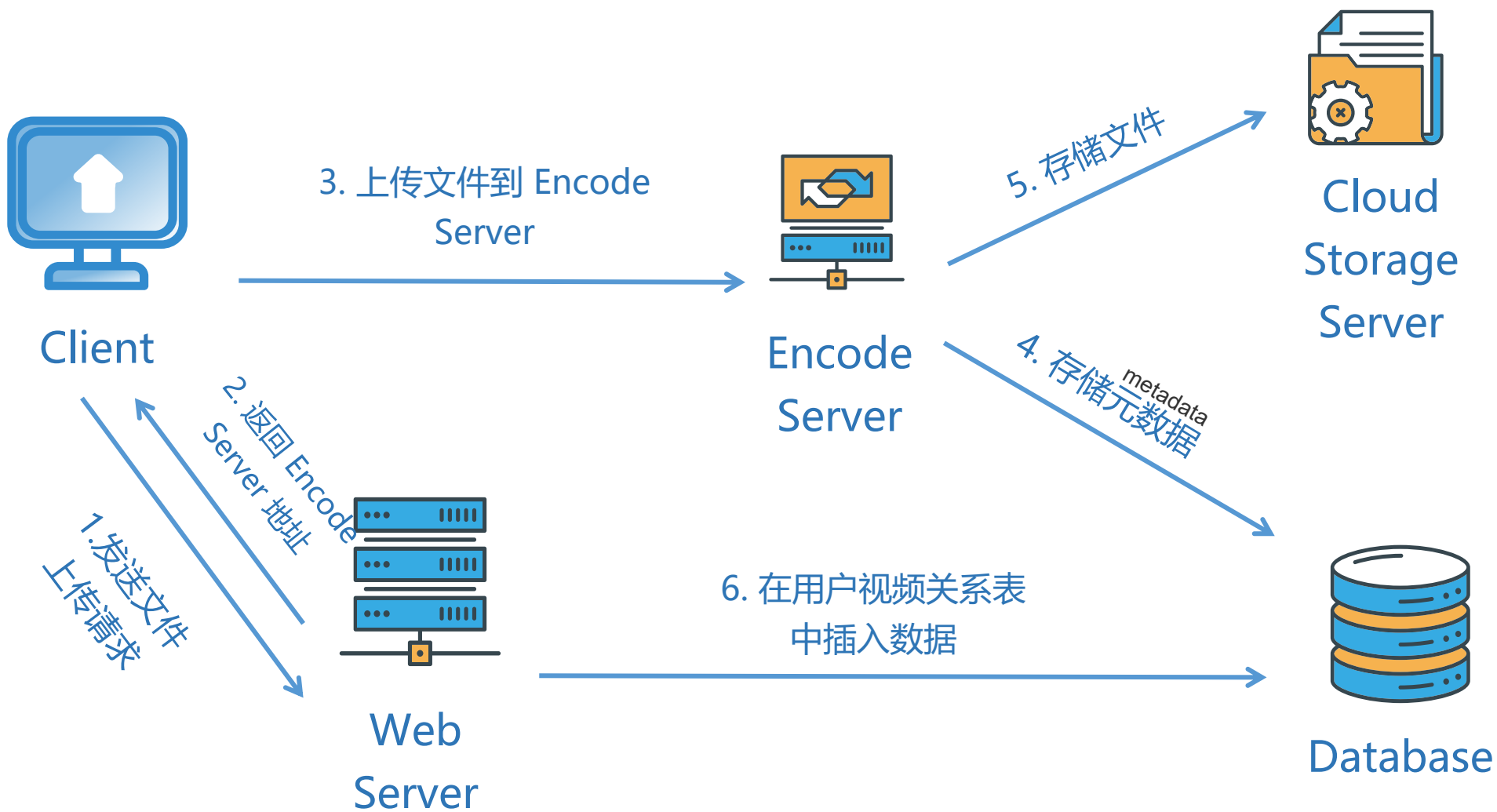


一对多：一个班级有很多的学生，通过外键进行连接。




多对多：一个学生有多门课程，一门课程有多名学生，通过一张中间表进行连接。

一个完整的文件上传流程



当用户上传重复的视频时会发生什么？

1. 两个人都上传到后端，并且存入文件系统，都在前端显示上传成功
2. 两个人都上传到后端，先上传的存入后上传的删除，都在前端显示上传成功 
3. 先上传的上传到后端，后上传的不上传，都在前端显示上传成功

前端不可信（会被劫持），必须在后端进行处理。

哪些 Cloud Storage Server 适合保存 Chunks 和缩略图?

- seaweedFS
- FastFS

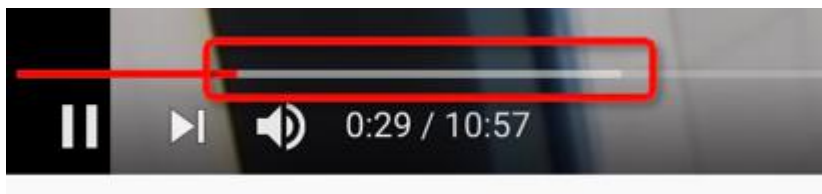
小文件存储和普通文件存储有什么区别？

- Chunks和图片Size一般偏小
- 读取频率较高，点击YouTube先看到的都是^{thumbnail}缩略图

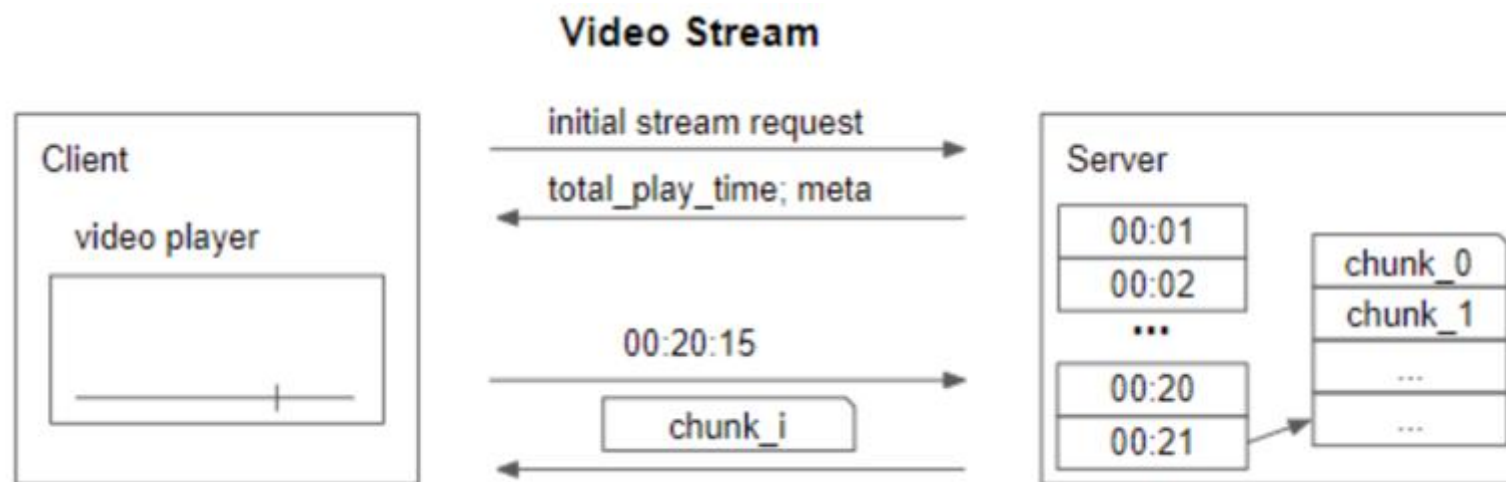
指标	适合类型	文件分布	系统性能	复杂度	FUSE	POSIX	备份机制	通讯协议接口	社区支持	去重	开发语言
FastDFS	4KB~500MB	小文件合并存储不分片处理	很高	简单	不支持	不支持	组内冗余备份	API/http	国内用户群		C语言
TFS	所有文件	小文件合并，以block组织分片		复杂	不支持	不支持	Block存储多份,主辅灾备	API/http	少		C++
MFS	大于64K	分片存储	Master占内存多		支持	支持	多点备份动态冗余	使用fuse挂在	较多		Perl
HDFS	大文件	大文件分片分块存储		简单	支持	支持	多副本	原生api	较多		java
Ceph	对象文件块	OSD一主多从		复杂	支持	支持	多副本	原生api	较少		C++
MogileFS	海量小图片		高	复杂	可以支持	不支持	动态冗余	原生api	文档少		Perl
ClusterFS	大文件			简单	支持	支持	镜像		多		C

04 视频播放与 缩略图加载

如何提高观看视频的流畅度？



边看边加载



01

封面缩略图

在前端加载视频列表的时候，去文件服务上读取

02

进度条缩略图

- 用户将鼠标移动到进度条上时，加载缩略图到本地缓存。
- 用户点开某个视频的时候，将该视频的所有缩略图加载到本地缓存。

前后端配合，前端传递加载的进度条位置数据，交由后端处理，从缓存中取缩略数据，返回当前进度缩略图

Scale扩展

1. 如何用 CDN 优化读取
2. 如何对数据库做 sharding

下面这串url访问文件有什么问题？

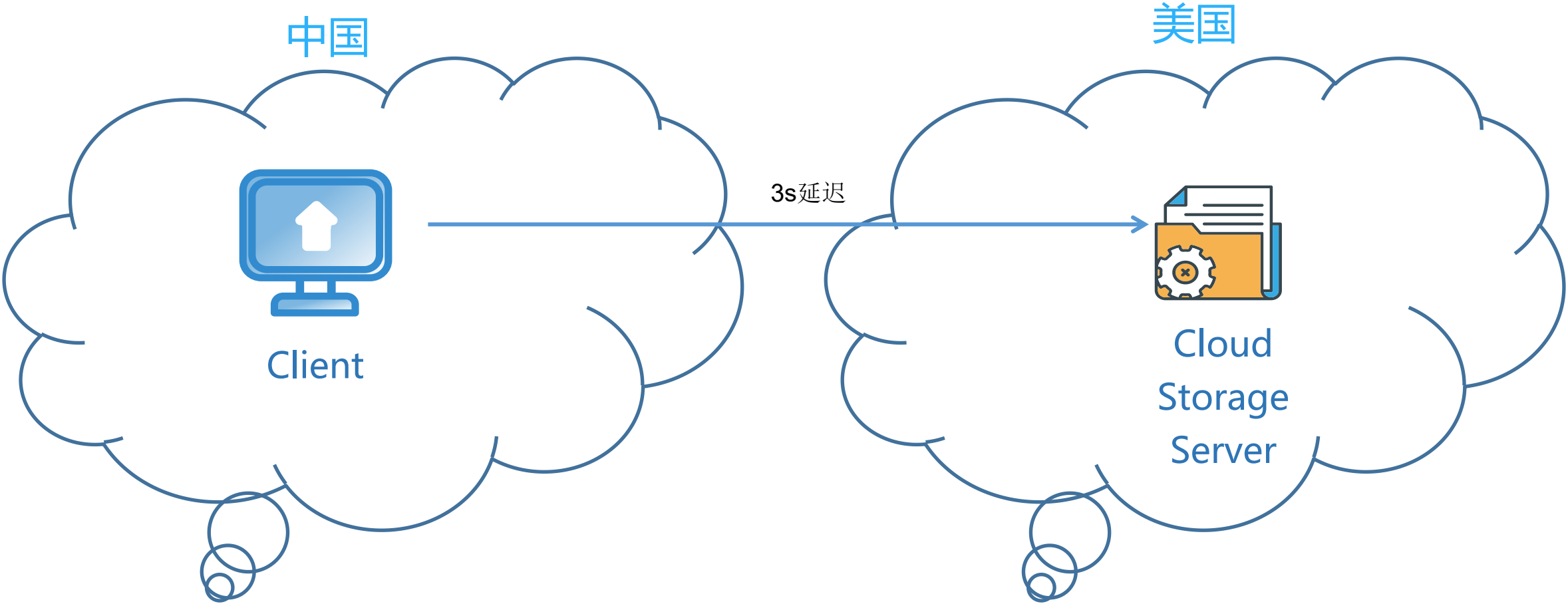
<https://www.jiuzhang.com/img/video-streaming.jpg>

将文件与webserver存放在了一起，破坏了 Webserver 的 stateless

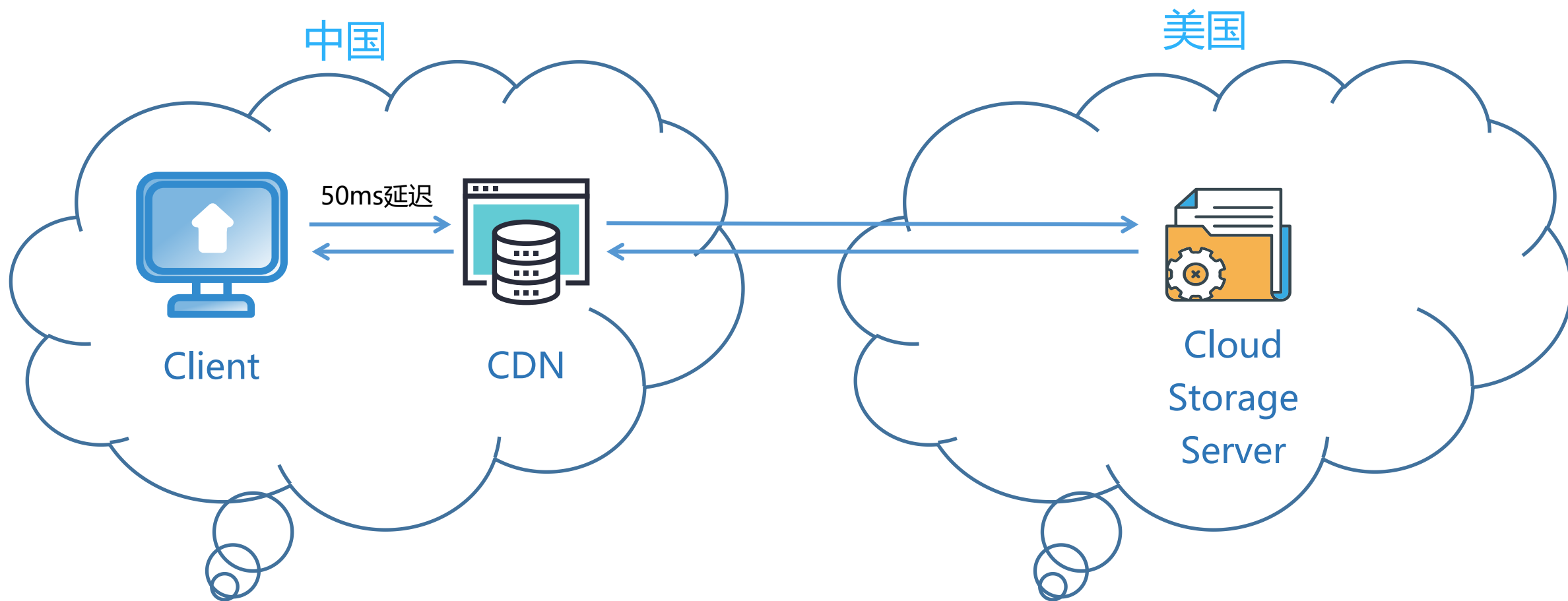
如何进行修改?

<https://www.jiuzhangStorage.com/img/video-streaming.jpg>

独立出一个文件系统来存储文件，将文件与 webserver 独立。

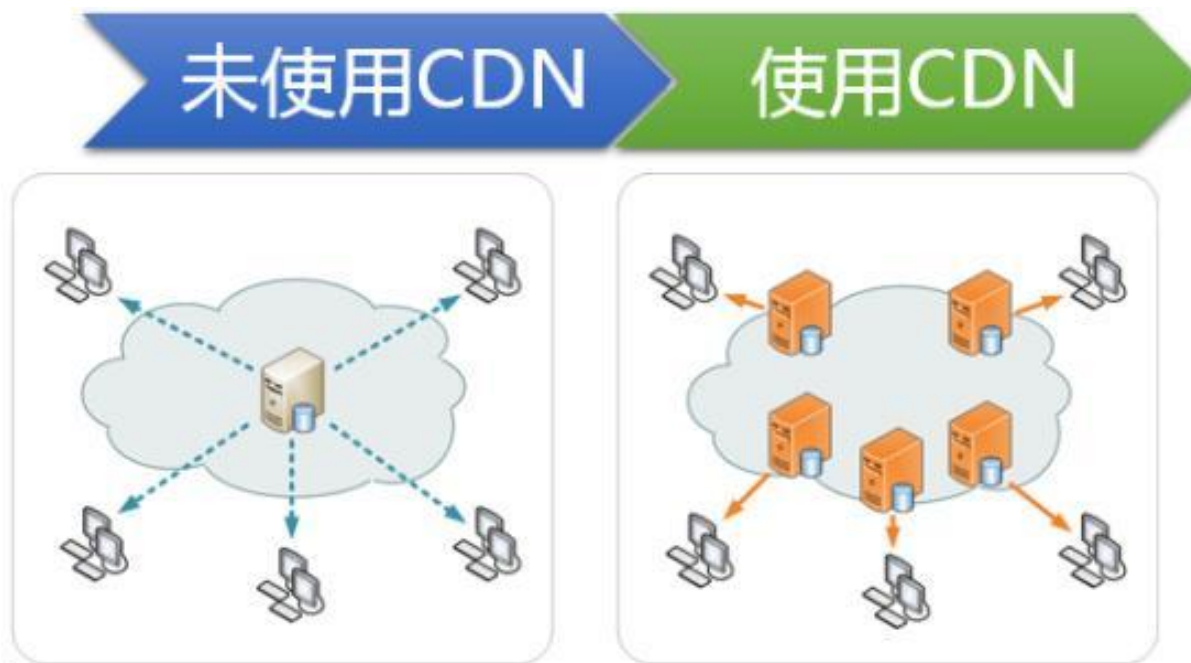


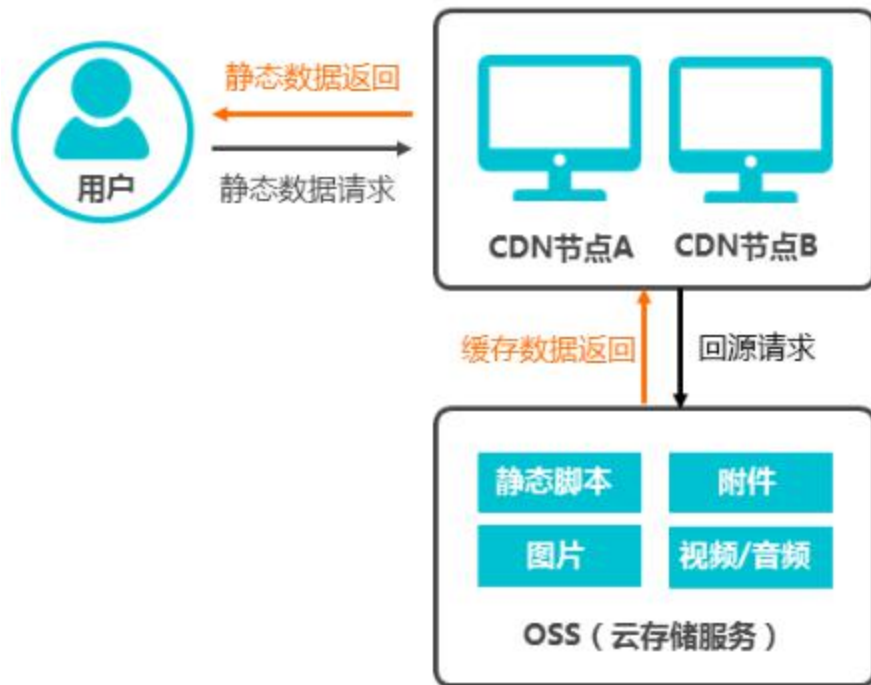
CDN(Content Delivery Network)



什么是CDN?

CDN是接近用户地理位置的边缘服务器，可以作为缓存处理



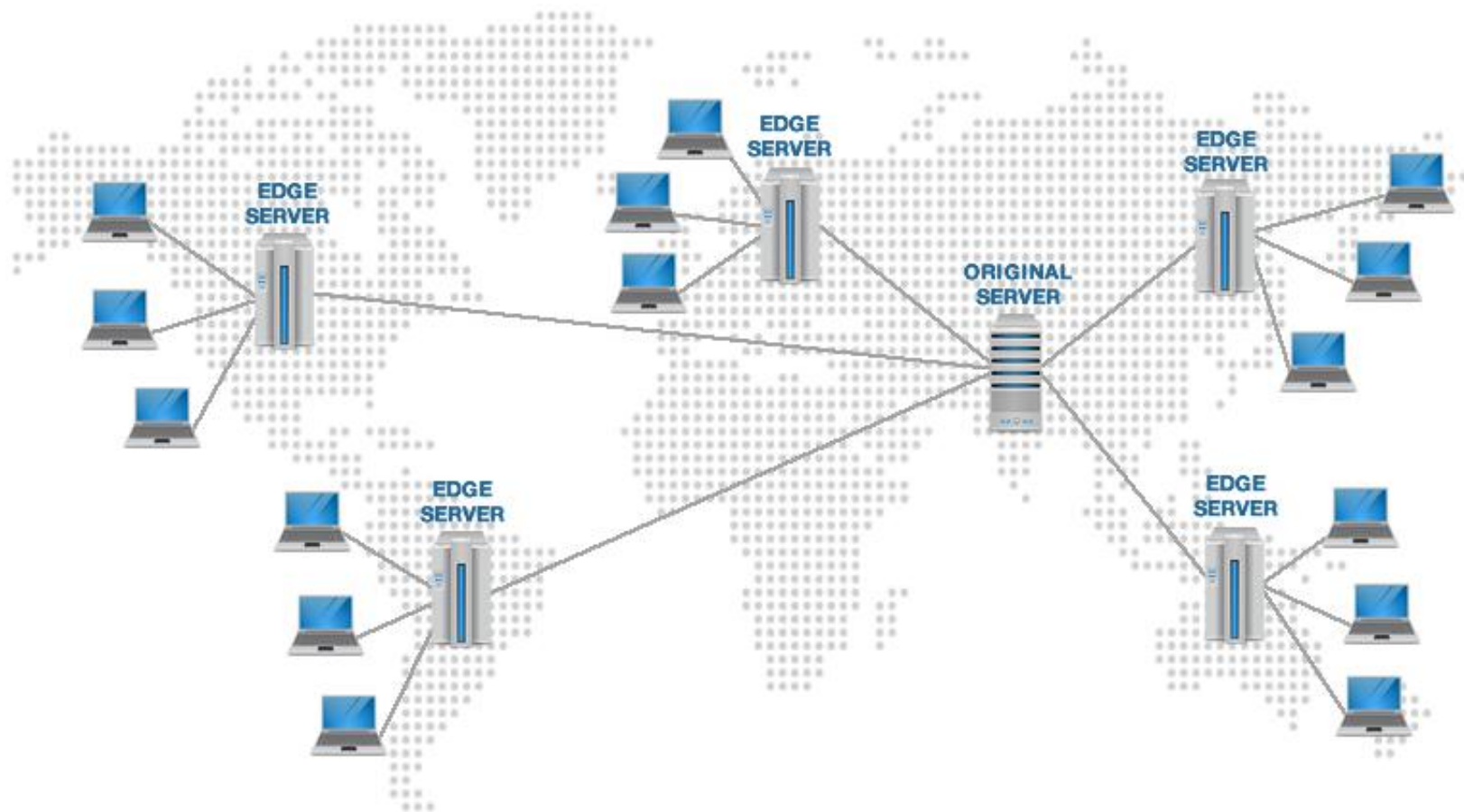


阿里云CDN官方文档

<https://help.aliyun.com/product/27099.html?spm=a2c4g.11186623.6.540.1e99e9c3rYIFhU>

CDN(Content Delivery Network)

- 80%的流量都是20%的视频贡献的
- YouTube会选择把**热点**视频定时同步到CDN
- 用户可以在就近的服务器访问到热点视频，无须请求中央服务器，加快加载速度



用户数据量越来越大怎么办?

对数据库进行 sharding

如何给数据库做 Sharding?

查询用户数据：用户视频关系表以用户ID做Sharding

查询视频数据：视频表、Chunk表以视频ID做Sharding

Thanks



下课啦，同学们来填一下问卷调查吧！
填完后截图给班班，有**200学分**哦！