Problem Statement

Design a video streaming platform

Youtube we can watch videos, upload videos on demand.

# Features

Scope of functional requirements (F.R)

1)Watch a video on demand.

2)Feature to watch the video from the time last watched.

3)Search for a video (keyword-based search)

4)Upload a video

5)View and add comments, likes

Scope of non-functional requirements (N.F.R)

1)Scalable

2)Available

3)Reliable/Trustworthy (If we upload a video, it should be ensured to stay all the time)

4)Latency should be low (from the time we searched the video and it started streaming)

# Estimation

How much storage, compute, network needed?

Total no of users on youtube ~ 2 B

50% are active users

No of active users ~ 1 B

Youtube has 2 kinds of traffic

i)Viewing/streaming videos

ii)Uploading videos

No of views = 400 x no of writes/uploads

On avg each user watches 4 videos per day.

Total no of videos watched per day = 1 B x 4 = 4 B

Total no of videos watched per second = 4 B/ (24 x 60 x 60)

~ 50 k videos per second

Now no of writes/uploads = 50 k / 400 ~ 125 uploads per second.

Storage needed

Depends on the how much of the size of the video that I am going to upload.

Let us say every minute we are uploading 400 hour worth of video data

1 minute of video data ~ 100 MB

400 hr => 400 x 60 x 100 MB ~ 2400 GB

For each second 2400/60 = 40 GB

So, we need 40 GB per every second.

# High level Design

Uploading

This should be done in the asynchronous fashion, so that the client does not need to wait till it gets uploaded.

So, that cleint should be able to check the status of the uploaded video.

As soon as user uploaded the content, there should be some kind of the video processing queue.(kafka)

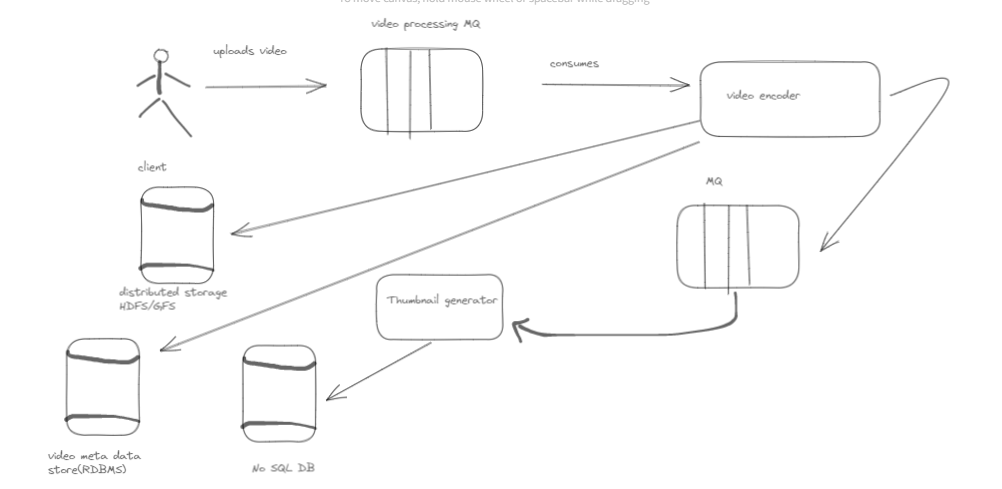
We need to encode the video to different video formats to support for the different formats.

After uploading is finished, we need to have some kind of thumbnails or the metadata.

As the rate of the storage is 40 GB / sec, we should need some kind of the distributed storage which practically does not have any upper limit like {HDFS, GFS}

When we store the video in the HDFS, we can store the meta data in some RDBMS which is consistent and can be indexed,it will contain the information like video name , tags , keywords , comments,likes,

For a given user on a given device what is the offset of video(till how he has watched)



Viewing

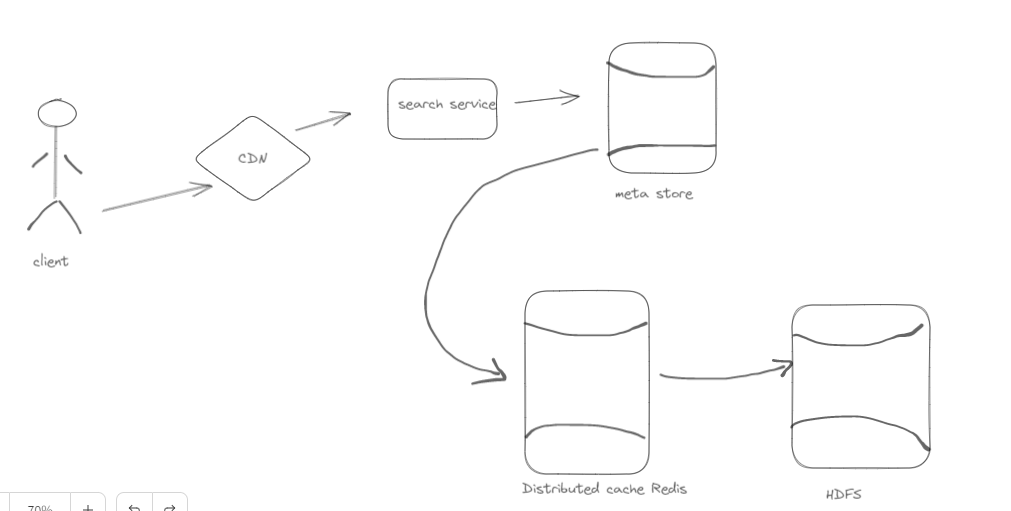
Search for a video.

Start watching the video.

Maintain the off set of the video based on the user and the device.

Usually when user try to search for the content from the HDFS directly it is very difficult to fetch and very slow operation also.

So , we should first go to meta data store which will tell if the video exists or not and index it quickly.



We can also place CDN which are third party cache providers.