DESIGN YOUTUBE / NETFLIX (video streaming platform)

NFR

→ No buffering (min. buffering possible)
(video should always be available and must serve using low latency)

→ User's session time

→ good recommendation engine, suggest the right content.

FR

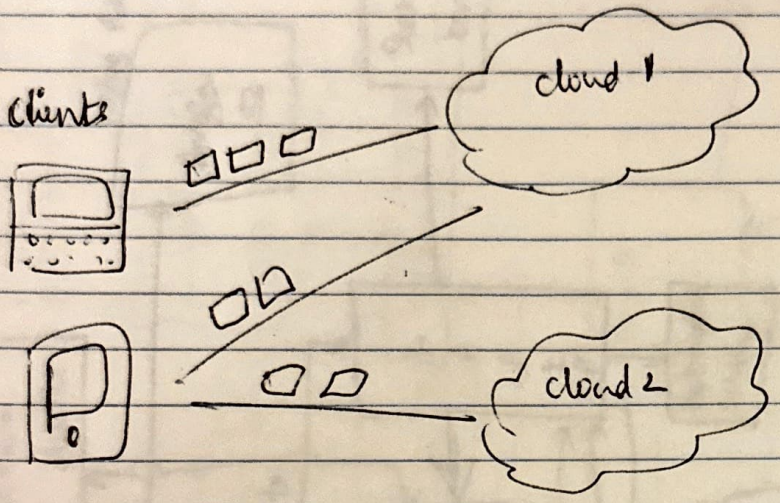
→ Upload videos
→ User's homepage + search
→ Play videos
→ Support all devices

Devices	Formats (i)	Dimensions (j)	Bandwidth (k)
Iphone			
Android			
iPad			
Laptops			
TVs			

'i' 'j' 'k' → These many ways to handle

Users are :-

1. Client (device)
2. User
3. Production houses (users who upload video).



~~Also~~ How this video streaming platform work is :-

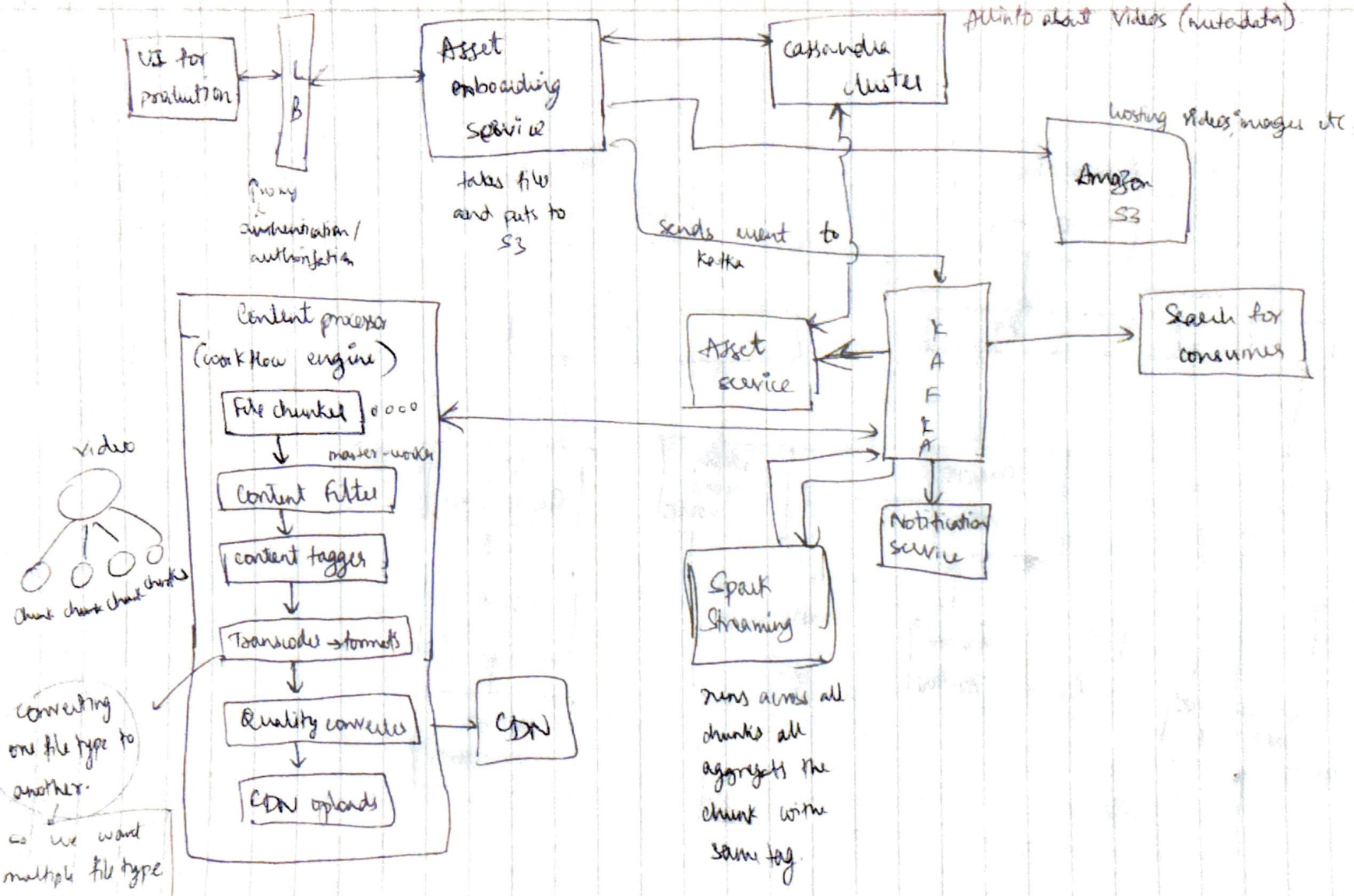
- They dont download the whole file to the device and start playing
- They request a chunk of the video play that, and while it is playing the video, they request for further chunks.
- client first figures out what device I am on, and what file formats I can support.

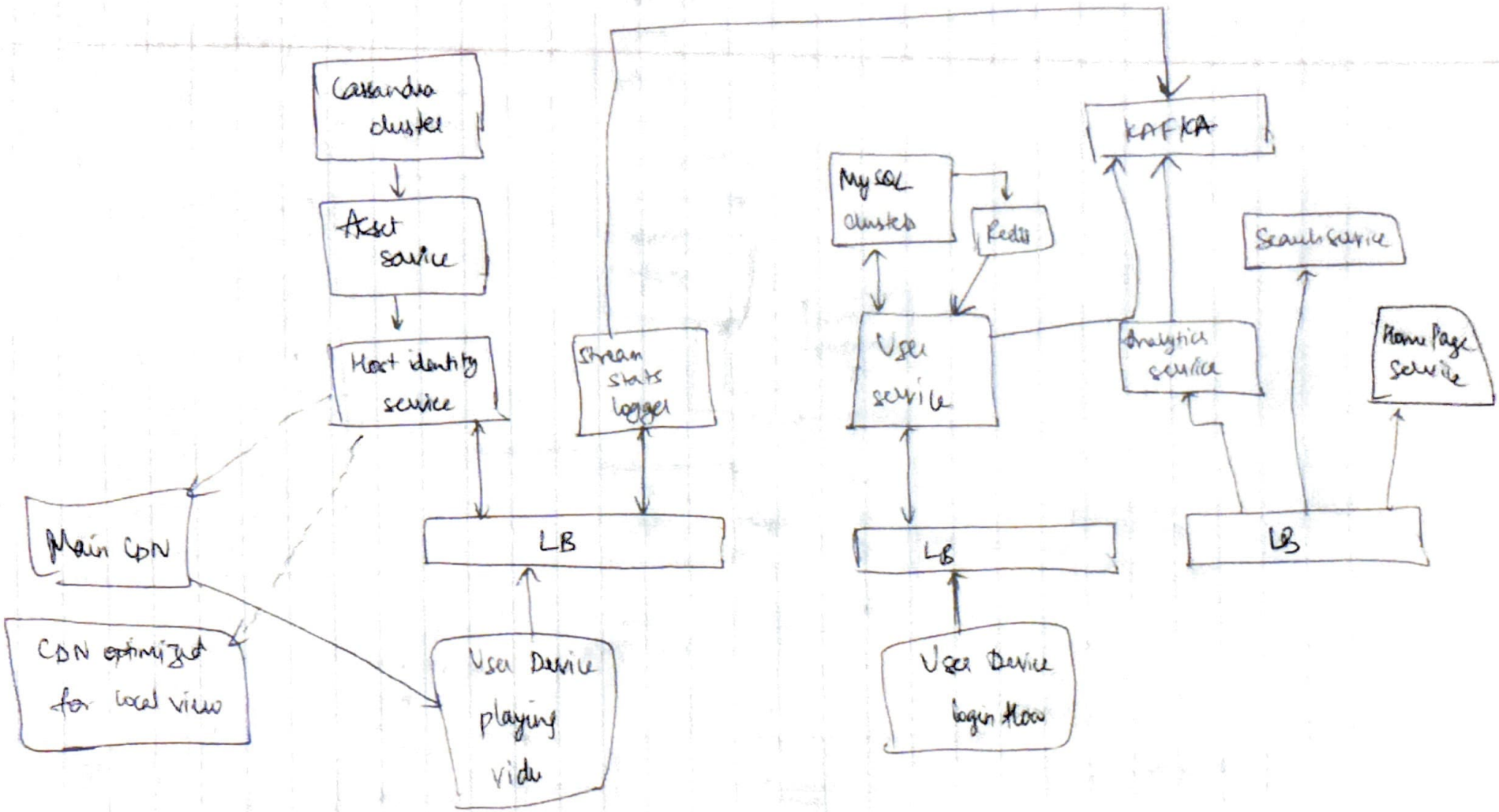
- And say we are playing a 1080 resolution video, but we get to know that the chunks are coming slowly, then if we get to know that the subsequent chunks are having low latency, then we request for a low resolution video for subsequent chunks. So, that we get all chunks in time.

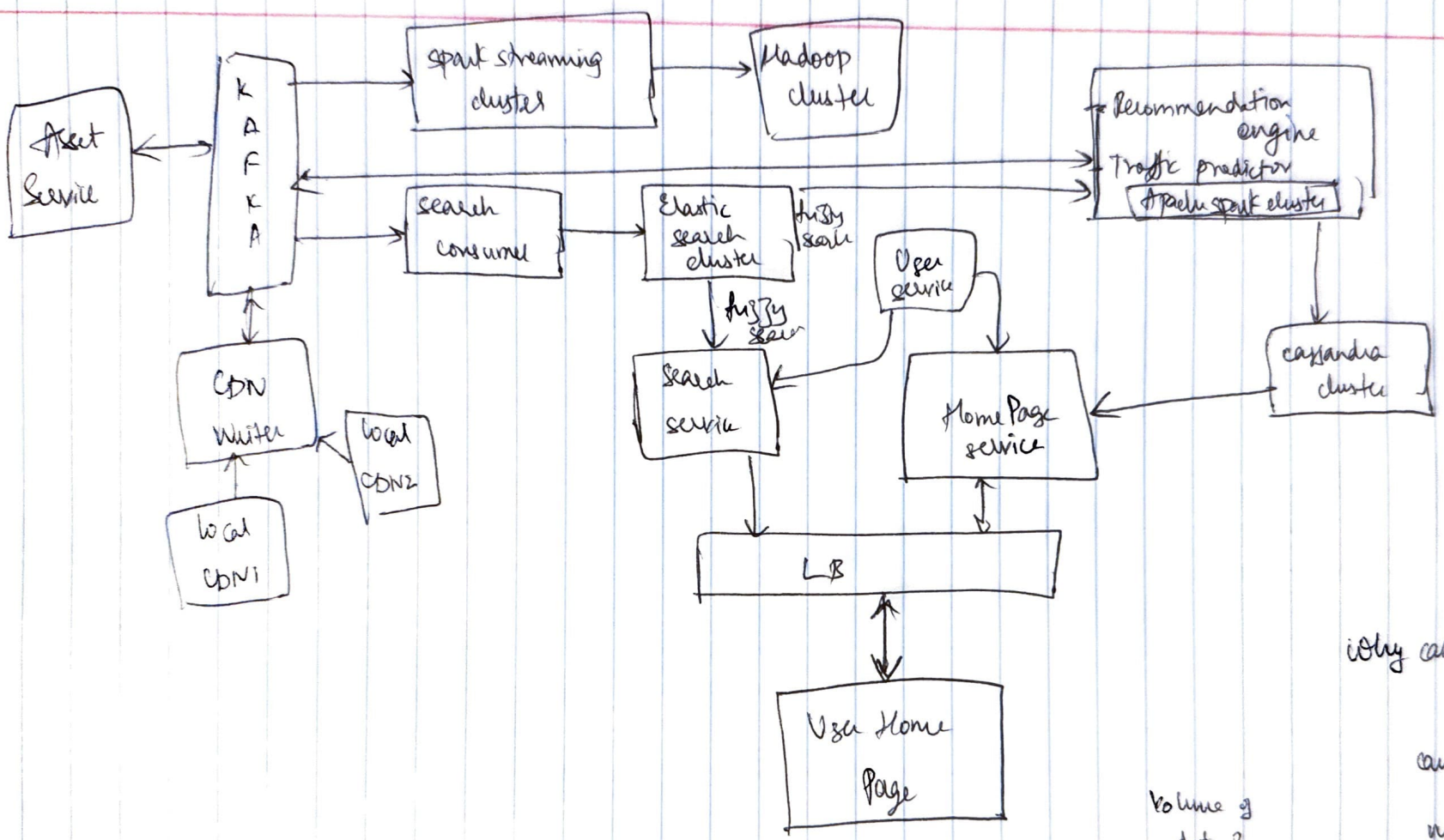
so basically ;

- So, at runtime it figures out like what bandwidth I have and adjust the quality of the video accordingly.

→ This is called adaptive bitrate streaming
(to improve user experience).


UI for production
LB
proxy
authentication / authorization
Asset onboarding service
takes file and puts to S3
Cassandra cluster
All info about Videos (metadata)
hosting videos, images etc
Amazon S3
sends event to Kafka
Asset service
K A F K A
Search for consumers
Content processor (workflow engine)
File chunker
master-worker
Content Filter
content tagger
Transcode → formats
Quality converter
CDN uploads
CDN
Video
chunk chunk chunk chunk
converting one file type to another.
so we want multiple file type as per requirements right
Spark Streaming
runs across all chunks all aggregate the chunk with same tag
Notification service


Cassandra cluster
Asset service
Host identity service
Stream stats logger
KAFKA
MySQL cluster
Redis
User service
Analytics service
Search service
Home Page Service
Main CDN
CDN optimized for local view
LB
User Device playing vidu
LB
User Device login flow
LB



Why cassandra ?

↓

can handly

millions of reads

and

millions of writes.

Volume of data ?

More than

billion videos