

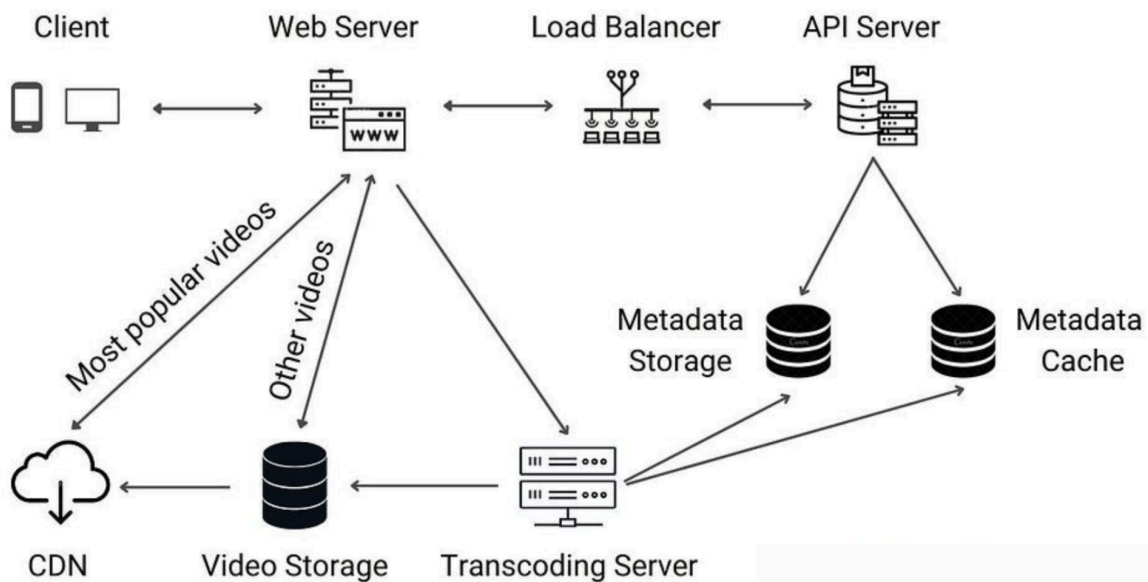
# Youtube

## Step 1

- Functional
  - Video **Watching** and **Uploading**
- Non-Functional
  - Watching : Uploading Ratio = 9 : 1
  - DAU: 1 Million
  - QPS:  $10^6 * 10 \text{ (requests)} / 86400 * 3 = 350$
  - Database: BLOB + SQL
  - Response Time: < 1s
  - Reliability Time: 99.9%

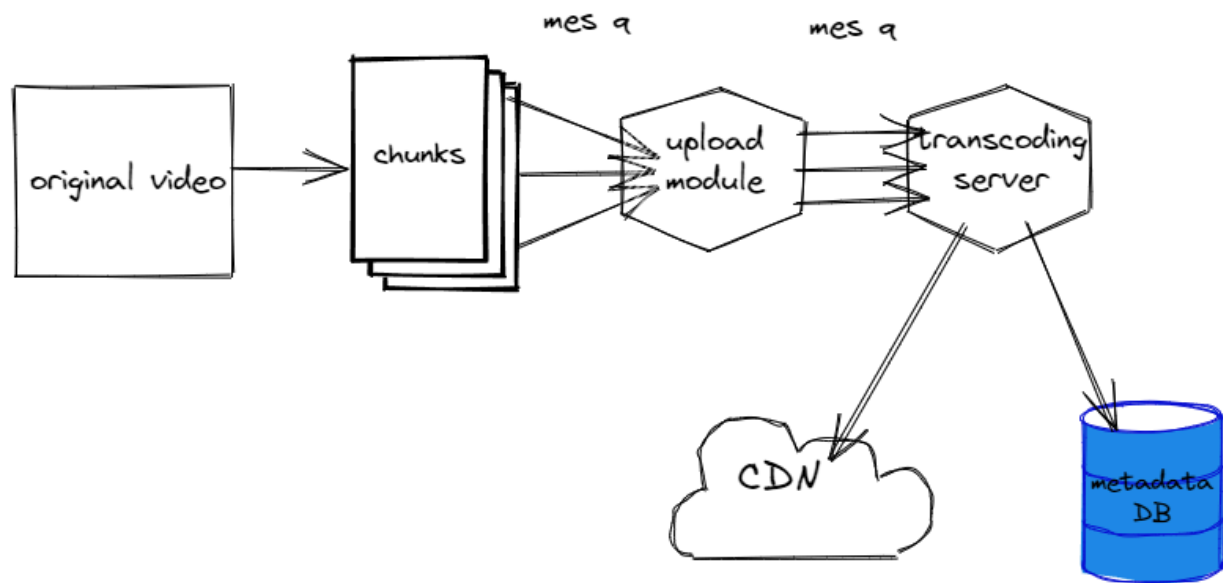
## Step 2

- Performance vs. Scalability
- Latency vs Throughput
- Consistency: Eventual
- Availability: Leader-Follower

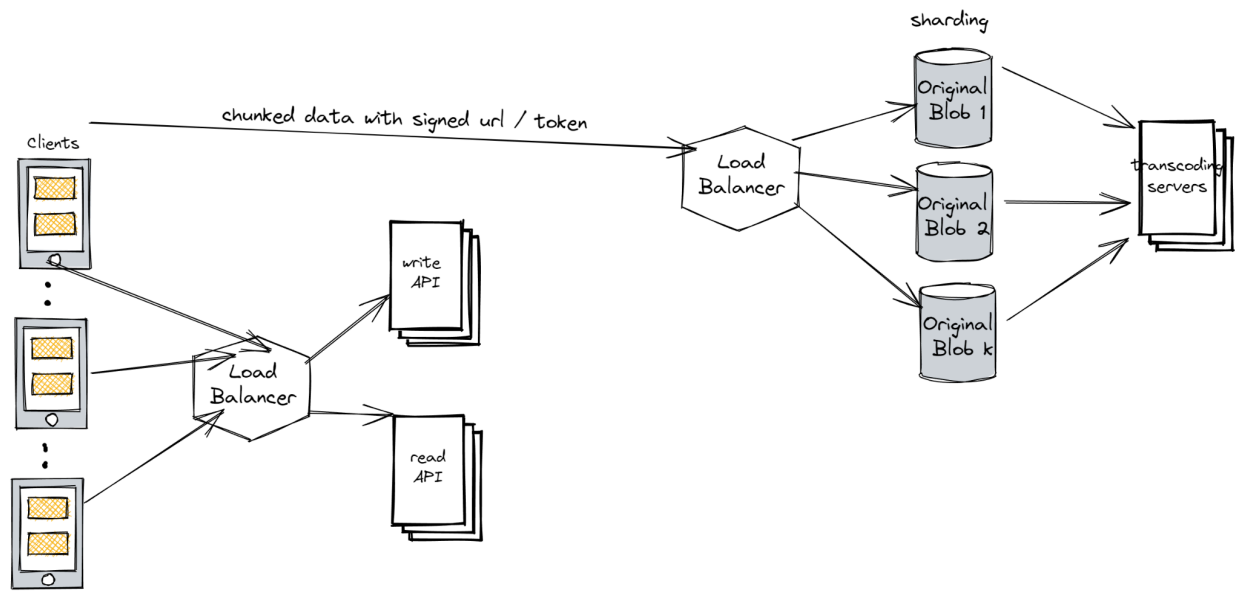


## Step 3

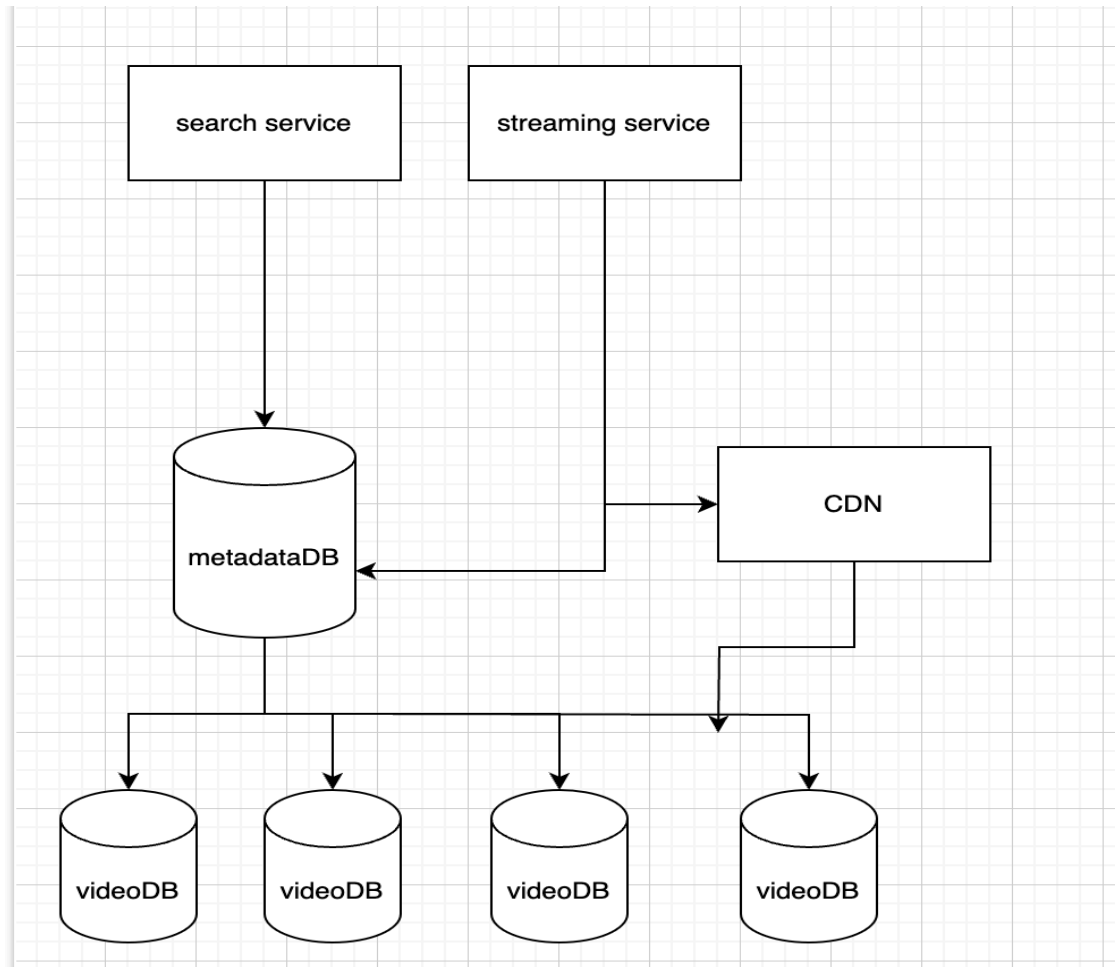
Transcoder for uploading



Uploading error handling, blob scaling



video searching and watching



## Step 4 scaling

## Appendix

### Schma设计

#### 用户表(User)

- UserID (主键)

- UserName
- Email
- PasswordHash
- SignUpDate
- LastLogin
- ProfilePictureURL

#### 视频表(Video)

- VideoID (主键)
- UserID (外键, 关联用户表)
- Title
- Description
- UploadDate
- FilePath (存储在对象存储的路径)
- ThumbnailURL
- Duration
- ViewCount
- LikeCount
- PrivacyStatus (公开、私密、不列出)

#### 评论表(Comment)

- CommentID (主键)
- VideoID (外键, 关联视频表)
- UserID (外键, 关联用户表)
- Text
- CommentDate

#### 标签表(Tag)

- TagID (主键)
- TagName

#### 视频标签关联表(VideoTag)

- VideoID (外键, 关联视频表)
- TagID (外键, 关联标签表)

#### 视频播放列表(Playlist)

- PlaylistID (主键)
- UserID (外键, 关联用户表)
- Title

- Description
- CreationDate

#### 播放列表视频关联表(PlaylistVideo)

- PlaylistID (外键, 关联播放列表)
- VideoID (外键, 关联视频表)
- AddDate