

# System Design

**Venkatesh Vinayakarao**

venkateshv@cmi.ac.in

<http://vvtesh.co.in>

---

Chennai Mathematical Institute

---

As an architect, you design for the present, with an awareness of the past, for a future which is essentially unknown. – **Norman Foster, a British Architect.**

# Define the Scope

- Users post and read short messages
- Messages may contain photos and videos
- Show trending topics

# Non-Functional Requirements

- Scalability
- High Availability
- Fault Tolerance
- Reliability
- Security
- Cost-effectiveness
- ...

# Capacity Estimation

- This involves determining the anticipated load or demand that a system will need to manage over a specified period.
- This encompasses estimating:
  - user count,
  - transactions,
  - storage capacity,
  - bandwidth, among other factors.

# Capacity Estimation – The Need

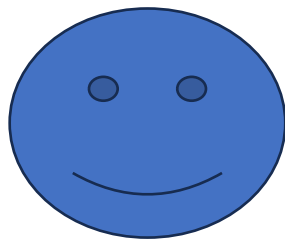
- Prevents
  - Over-provisioning
  - Under-provisioning
- Helps to ensure that the non-functional requirements are met.
- Determines architectural choices and tradeoffs.

# Capacity Estimation - Assumptions

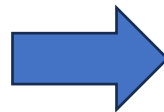
**1 Message every 2 days**

**1 Image (200 KB) in every 5th Message**

**1 Video (2 MB) in every 10th Message**



**200 Million  
Daily Active Users**



**140 Characters**

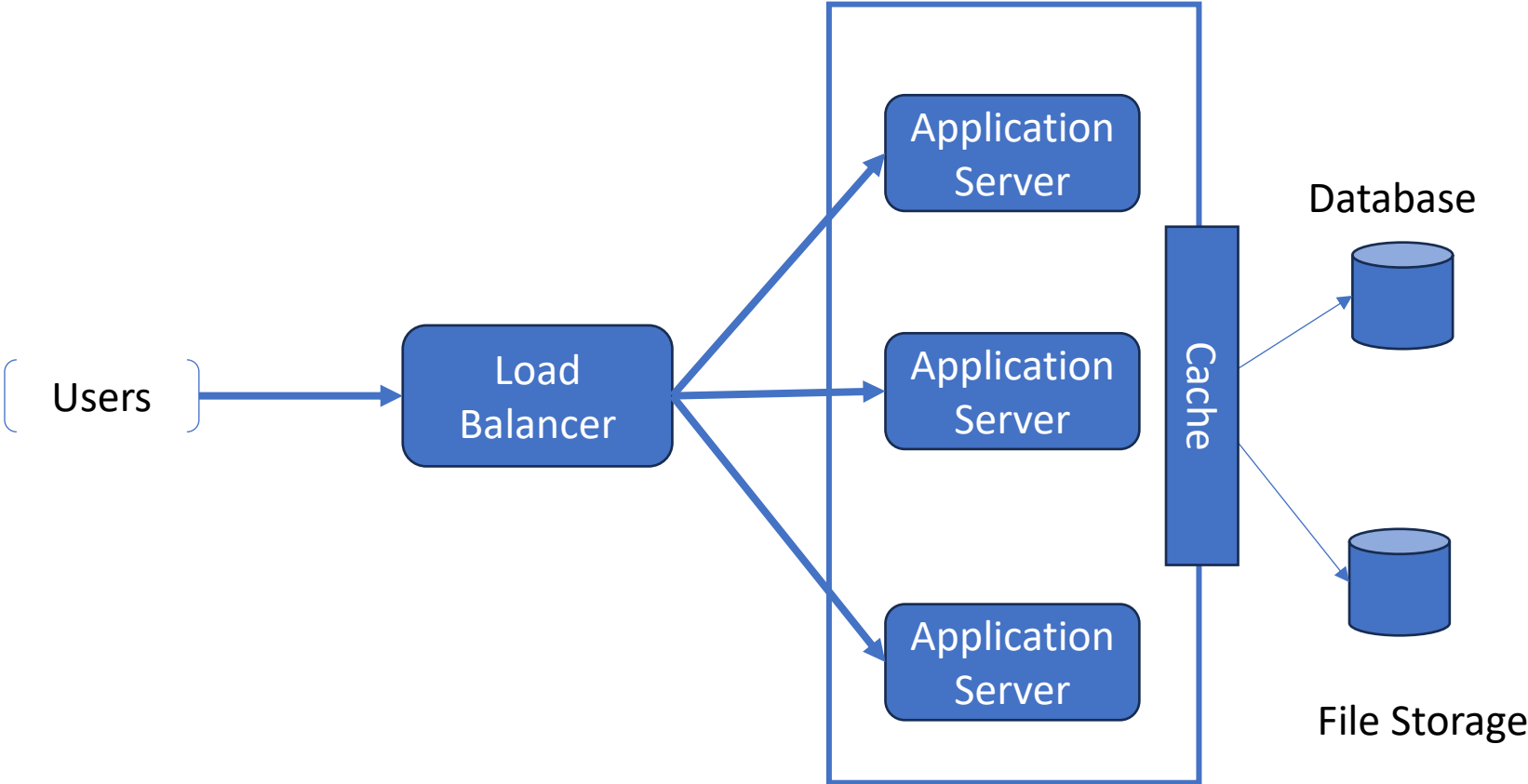


**Metadata of 30 bytes**  
(datetime, user, etc)

# Capacity Estimation

- How many messages are created per day?
  - 100M messages per day
- How much space (in GB) will we need per day?
  - We need  $100M * 2 * (140 + 30)$  bytes = (approx.) 30 GB for text per day.
- Overall, how much data (text + images + videos) is generated per day?
  - 24 TB of data per day!

# System Design





# Which DB to use?

- Read heavy system.
- Eventual consistency is good enough.
- We prioritize availability and partition tolerance.
  - So, Cassandra is a good choice.

**How to show trending  
topics?**

# How to Show Trending Topics?

- We can use hashtags in the messages.
- Cache hashtags and their count that were used in the recent N minutes.
- Update the cache frequently.
  - Use a batch job.

Thank You