



# Introduction to Pub-sub

Learn about the use cases of the pub-sub system, how to define its requirements, and design the API for it.

## We'll cover the following




- Use cases of pub-sub
- Requirements
  - Functional requirements
  - Non-functional requirements
- API Design
- Building blocks we will use

Pub-sub messaging offers asynchronous communication. Let's explore the use cases where it is beneficial to have a pub-sub system.

## Use cases of pub-sub

A few use cases of pub-sub are listed below:

- **Improved performance:** The pub-sub system enables push-based distribution, alleviating the need for message recipients to check for new information and changes regularly. It encourages faster response times and lowers the delivery latency.
- **Handling ingestion:** The pub-sub helps in handling log ingestion. The user-interaction data can help us figure out useful analyses about the behavior of users. We can ingest a large amount of data to the pub-sub system, so much so that it can deliver the data to any analytical system to understand the behavior patterns of users. Moreover, we can also log the details of the event that's happening while completing a request from the user. Large services like Meta use a pub-sub system called Scribe to know exactly who needs what data, and remove or archive processed or unwanted data. Doing this is necessary to manage an enormous amount of data.

- 
- **Real-time monitoring:** Raw or processed messages of an application or system can be provided to multiple applications to monitor a system in real time.
  - **Replicating data:** The pub-sub system can be used to distribute changes. For example, in a leader-follower protocol, the leader sends the changes to its followers via a pub-sub system. It allows followers to update their data asynchronously. The distributed caches can also refresh themselves by receiving the modifications asynchronously. Along the same lines, applications like WhatsApp that allow multiple views of the same conversation—for example, on a mobile phone and a computer's browser—can elegantly work using a pub-sub, where multiple views can act either as a publisher or a subscriber.

