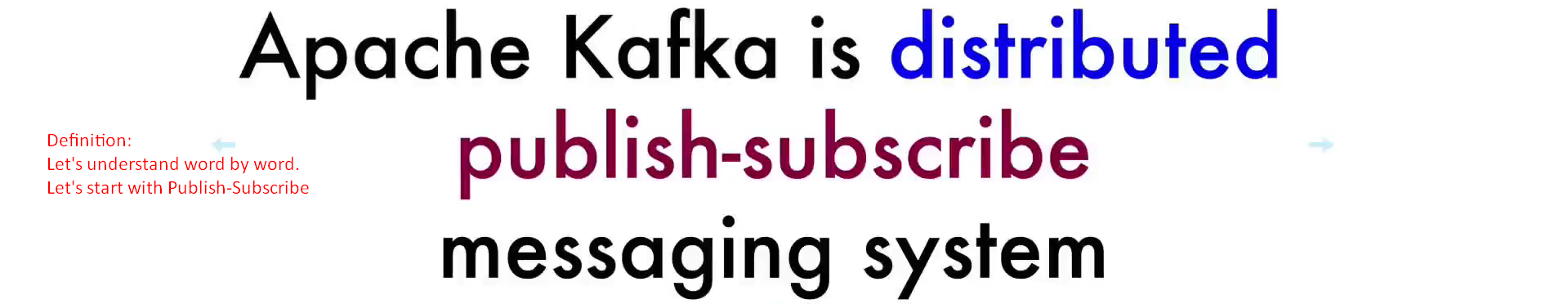
1. Graphical user interface, text

   Description automatically generated
2. 
3. **Publish Subscriber System**:
   1. It means there are publishers and there are subscribers.
   2. **Publishers**: They publish some information.
   3. **Subscribers**: They subscribe to that information.
   4. **Example**:
      1. **Youtube** is such a system.
      2. There are **creators** on Youtube who publish them at Youtube platform at any **moment of time**.
      3. On the other side, there are consumers who are able to watch any video at any moment of time.
      4. Youtube as Pub-Sub system stores every single video as msg.
      5. Publishers can publish video any moment of time and don’t know when consumers/watchers are going to watch.
      6. So, Youtube as centralized system is responsible for storage of all videos and make them available for watchers.

On the other hand, watchers/consumers are independent from the publishers.

1. Next word in the definition is **“Messaging”**
   1. which means Publishers and Subscribers exchange only messages not videos or other things.
   2. Every message is a sequence of bytes nothing else.
   3. The responsibility of Apache Kafka is to store those messages published by Publishers and deliver those messages to Subscribers whenever they ask for them.
2. Next word in the definition is **“Distributed”**.
   1. It means Apache Kafka is **Fault Tolerant** **Resilient System** with ability to create a large cluster with many servers that allows you to create a Fault Tolerant System.   
      Whenever if any of the servers gets failed, others will keep on serving Publishers & Consumers.
   2. If everything is set up correctly, even single msg will not be lost.
3. In next lecture, let’s discuss about **Broker**.