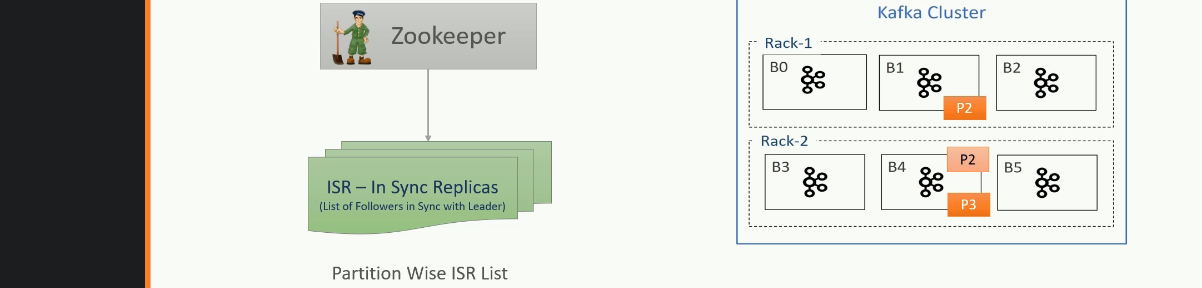
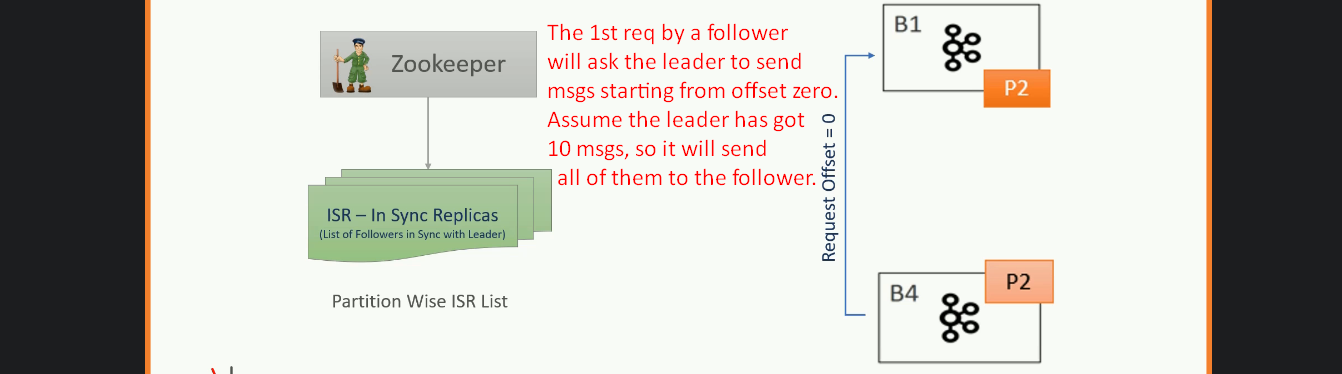
1. Text

   Description automatically generated
2. 
3. We already learnt in the some lecture that followers will continuously request for the msgs from the Leader Brokers to stay in sync with the leaders.
4. 
5. The method of copying msgs at the followers appears full-proof.
6. However, some followers may fail to stay in sync with leaders for various reasons.
7. Two common reasons are:
   1. **Network congestion**: May slow down the replication and so followers may start falling behind.
   2. **Follower Broker crashes/Restart**: When a follower broker crashes, all replicas on that broker will start falling behind until we restart the follower broker and they (replicas) will start replicating again.
8. Since the replicas may be falling behind, the leader has one more important job of maintaining a list of **I**n-**S**ync-**R**eplicas (ISR).
9. This list is known as the **ISR list of the partition** & **persisted in the Zookeeper and this list is maintained by the leader broker**.  
     
   The ISR list is very **critical**. **Why?**Because all the followers in that list are known to be in sync with the leader & they are excellent candidate to be elected as a new leader when something wrong happens to the current leader Broker. **That is why ISR is critical**.
10. However, there is one question that follows.  
    How does the leader know if the followers are in sync so should be included in the ISR or they are lagging and so it should be removed from ISR?
11. Let’s try to understand that.  
    **Jatin**: We will use some terms like
    1. **Leader Broker**: It means we are referring to leader broker having **leader partition**.
    2. **Follower Broker**: It means we are referring to follower broker having **follower partition** corresponding to above **leader partition**.  
       **For Example**: Follower broker will ask Leader broker to send msgs starting with offset=0.   
       It means follower broker has a follower partition say F1 and Leader Broker has the corresponding leader partition L1.   
       If a broker is asking another broker to send msgs it means the one asking is follower broker having follower partition and the one sending msgs is the leader broker having corresponding leader partition. 😊 That simple!!!
12. The follower will connect to the Leader and will ask for the msgs.  
    The 1st request will ask leader partition to send msgs starting with Offset=0.  
    Assume that the leader has 10 msgs and it will send all of them (offset=0-9) to the follower.  
    Follower will save into the respective replica and again ask for msgs starting offset=10.  
    And Leader will assume that all the msgs up to offset=9 are persisted into the follower.  
    So just by looking at the request offset requested by follower, the lead can decide whether a follower is in sync or not.

Now, ISR list is easy to maintain.  
If replica is not too far, the leader will add the follower to the ISR List else the follower is removed from the ISR list.

That means the ISR list is dynamic, and followers keep getting added and removed from the ISR list depending on how far they maintain their In-Syn Status.

1. 
2. Text

   Description automatically generated with low confidence  
   Chart

   Description automatically generated with medium confidence
3. However, there is a catch here?  
   How do we define the “**Not too Far**” as shown in the above diagram?
4. As a matter of fact, a follower will be always a little behind the leader and that is obvious because follower needs to ask for the msg from the leader, store them in the replica and ask for more.  
   All of these activities take some time and hence, the leader gives them some minimum time as a margin to accomplish this.  
   That is where the notation of “**Not too Far**” arrives.
5. The default value for “**Not too Far**” is 10 seconds.
6. But we can increase/decrease it using kafka configurations.
7. So, a replica can be kept in the ISR list if it is not more than 10 sec behind the leader.  
   That means if a replica (A broker with partition replica) has requested the most recent message in the last 10 seconds, they deserve to be in the ISR.  
   If not, the Leader removes the replica from the ISR (**I**n-**S**ync-**R**eplica).
8. Diagram

   Description automatically generated
9. Jatin: Basically, in this lecture we wanted to know:
   1. How to maintain a list of followers which are in syn with the leader so that in case the leader is down, we can choose a new leader from the list of those followers.
   2. We noticed that ISR is the solution maintaining the list of followers in syn with leader.
   3. If a follower makes a request for the last msg within 10 seconds, the follower is added to the ISR list.
   4. A follower already in the ISR list is removed.
      1. A close up of a text

         Description automatically generated  
         