1. Text

   Description automatically generated  
   
2. 
3. The mechanism to maintain ISR list is fancy.  
   However, we still have **gotcha** in this mechanism.
4. The maintenance of ISR list lead to some new concepts.   
   Let’s try to understand it.
5. We already understand that the followers may be lagging behind the Leader for several reasons and the followers might get removed from the ISR.
6. Diagram, schematic

   Description automatically generated
7. Diagram

   Description automatically generated
   1. Suppose all the followers in the P2-ISR are 11 sec behind the Leader.   
      That means none of them is qualified to be in the P2-ISR.  
      So, your P2-ISR list will become logically empty.  
      As per P2-ISR, the messages are only at the Leader B1 as P2-ISR has only B1 (Even though B2, B4 have some msgs but P2-ISR will be considered)
   2. Diagram

      Description automatically generated
   3. New for some reasons, the Leader B1 crashes and we need to elect a new Leader.  
      Who do we choose as a new Leader?  
      If we elect a Leader among those followers which were in the P2-ISR, will we not lose those messages that are collected at the leader during the most recent 11 seconds?  
      Yes we miss them.  
      But how do we can handle?  
      We don’t want to lose anything.
   4. The solution is very simple which is implemented using two concepts.  
      Text

      Description automatically generated with low confidence  
      The first idea is to introduce the notion of committed or uncommitted message.  
      The second idea is to set minimum in sync replica configuration.  
        
      Let’s try to understand these two things.
8. **Committed Vs. Uncommitted**:  
   Diagram

   Description automatically generated
   1. We can configure Leader not to consider a msg until the msg is copied to all the followers in the ISR list.
   2. If we do that, then the Leader may have some committed and some un-committed msgs.
   3. Now committed msgs we can’t lose it until we lose all the replicas.
   4. Now committed msgs are taken care of.
   5. However, we lose the leader itself then uncommitted msgs will be lost.
   6. But uncommitted msgs would not be a worry as those can be resent by the producer.
   7. Why?
   8. Because producer chooses to receive acknowledgement of sent msgs only after a msg is fully committed.  
      In that case, the producer waits for the acknowledgement for a timeout period and resends the msgs in the absence of commit acknowledgement.  
      So, the uncommitted msgs are lost at the failing Leader but the newly elected leader will receive those msgs again from the Producer.  
      That is how all the msgs can be protected from loss.
9. If idea is not clear. No issue as we will be covering it again in the course.