

**Definition: Replication**  
 Two independently and  
 regularly generated copies are  
 a "replica" of the data  
 distributed in two parts  
 of the system. The result  
 is that the data is not  
 lost if a part of the  
 system is down.

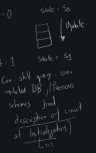
**Advantages:**  
 - In the event of a failure, the  
 system can continue to  
 operate with the other  
 copy.

**Disadvantages:**  
 - Consistency: updates  
 must be propagated to  
 all replicas.

**Three-Party Model**



**Three-Party Model**



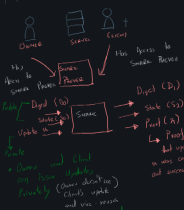
**Getting started**

**Three-Party Model**



It is a redundancy  
 system. The data is  
 stored in three places.  
 If one place fails, the  
 data is still available.  
 This is a good thing.  
 But it is also a bad  
 thing. If you have  
 three copies of the  
 data, you need to  
 update all three.  
 This is a problem.

**(Random Idea)**



But, this is not the  
 only way to do it.  
 There are other ways  
 to do it. For example,  
 you could use a  
 distributed system.  
 This is a good idea.

**Three-Party Model**

• Owner also gives the update  
 to the client.  
 • Client verifies that the object  
 is the same as the one  
 in the server. If not,  
 the client updates the  
 server.

**Novelty with Summary**

• Client sends a private  
 update to the server.  
 • Server sends a summary  
 of the update to the  
 client. The client  
 verifies that the  
 summary is correct.

**Intuition of Construction**

The goal is to construct a data structure that supports the following operations:

**Set Operations**

Union (A ∪ B)

Intersection (A ∩ B)

Set Difference (A - B)

Set Complement (U - A)

**Background**

• Subset

• Containment

• Disjoint

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

• Summary

- 1) Input: a set of elements
- 2) Output: a set of elements
- 3) Input: a set of elements
- 4) Output: a set of elements
- 5) Input: a set of elements
- 6) Output: a set of elements
- 7) Input: a set of elements
- 8) Output: a set of elements
- 9) Input: a set of elements
- 10) Output: a set of elements