Problem with trying to keep data in sync at 2 places (i.e. in a master data store & in cache)

Assume we have an item – x, which is present in both data base & cache with value = 1.  
Database & cache are present in 2 separate machines

* Client1 🡪 send request to update x = 2 at both database & cache
* R1DB 🡪 to update in database (x = 2)
* R1C 🡪 to update in cache (x = 2)
* Client2 🡪 send request to update x = 3 at both database & cache
* R2DB 🡪 to update in database (x = 3)
* R2C 🡪 to update in cache (x = 3)
* Assume if the requests are received in the following order R1DB, R2DB, R2C, R1C
* After both the clients request has been processed, the database & cache will be in in-consistent state, i.e. DB will have x = 3, while cache will have x = 2
* This problem exists as we are trying to modify at two places and both the operation is not in the same transaction. If we perform the set of operations (R1DB, R1C) in a transaction, i.e. either both R1DB & R1C succeed or fail together and (R2DB, R2C) in its own transaction then the data will be in consistent state
* Performing transaction on 2 separate machine requires 2 Phase commit which is an expensive operation