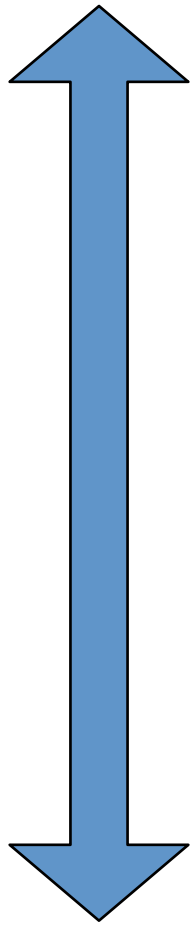
Large-Scale Data Processing

Yang Wang

Consistency

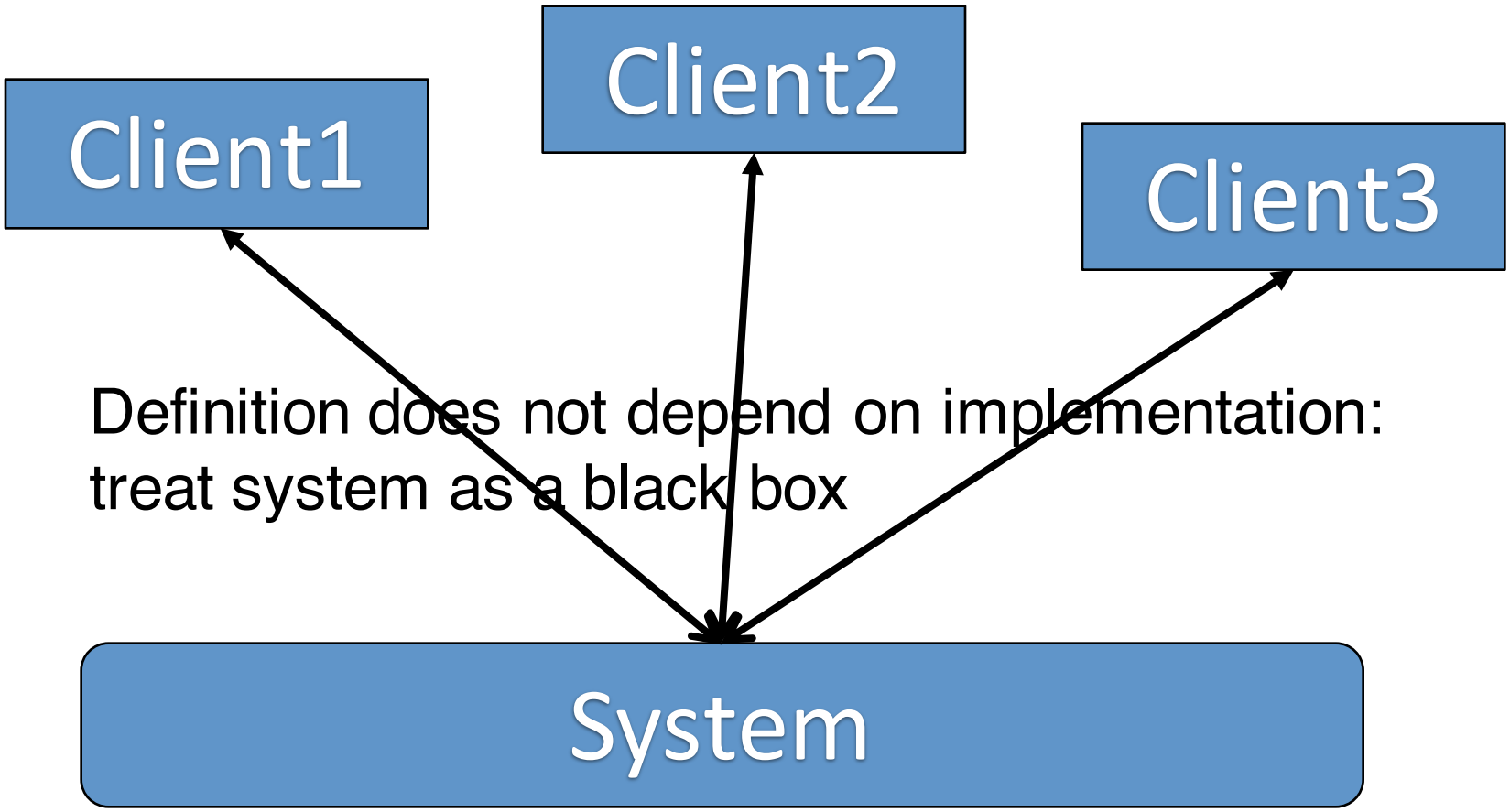
- Linearizable
- Sequential
- Causal
- Eventual



Strong

Weak

Consistency

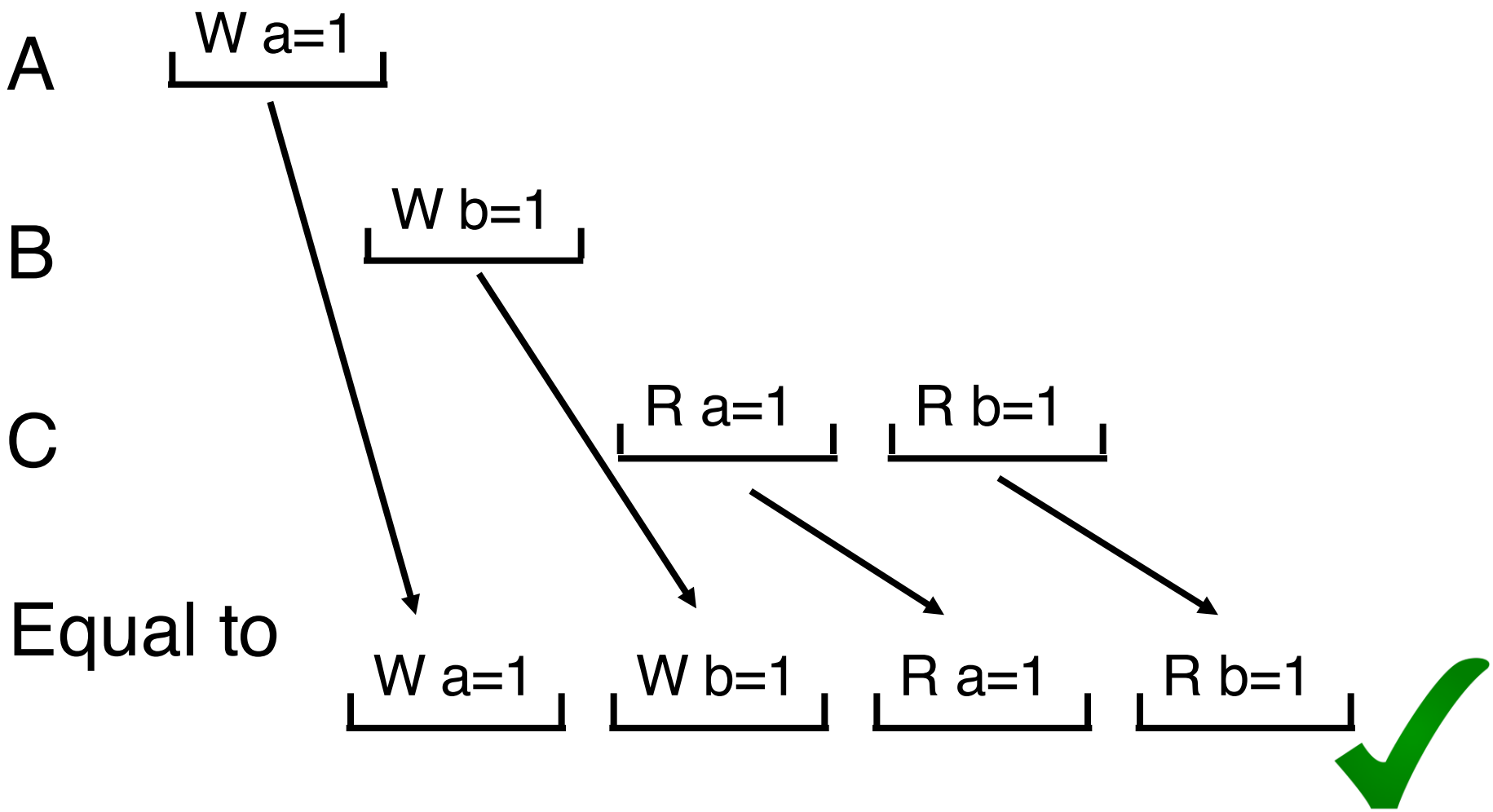


Definition does not depend on implementation:
treat system as a black box

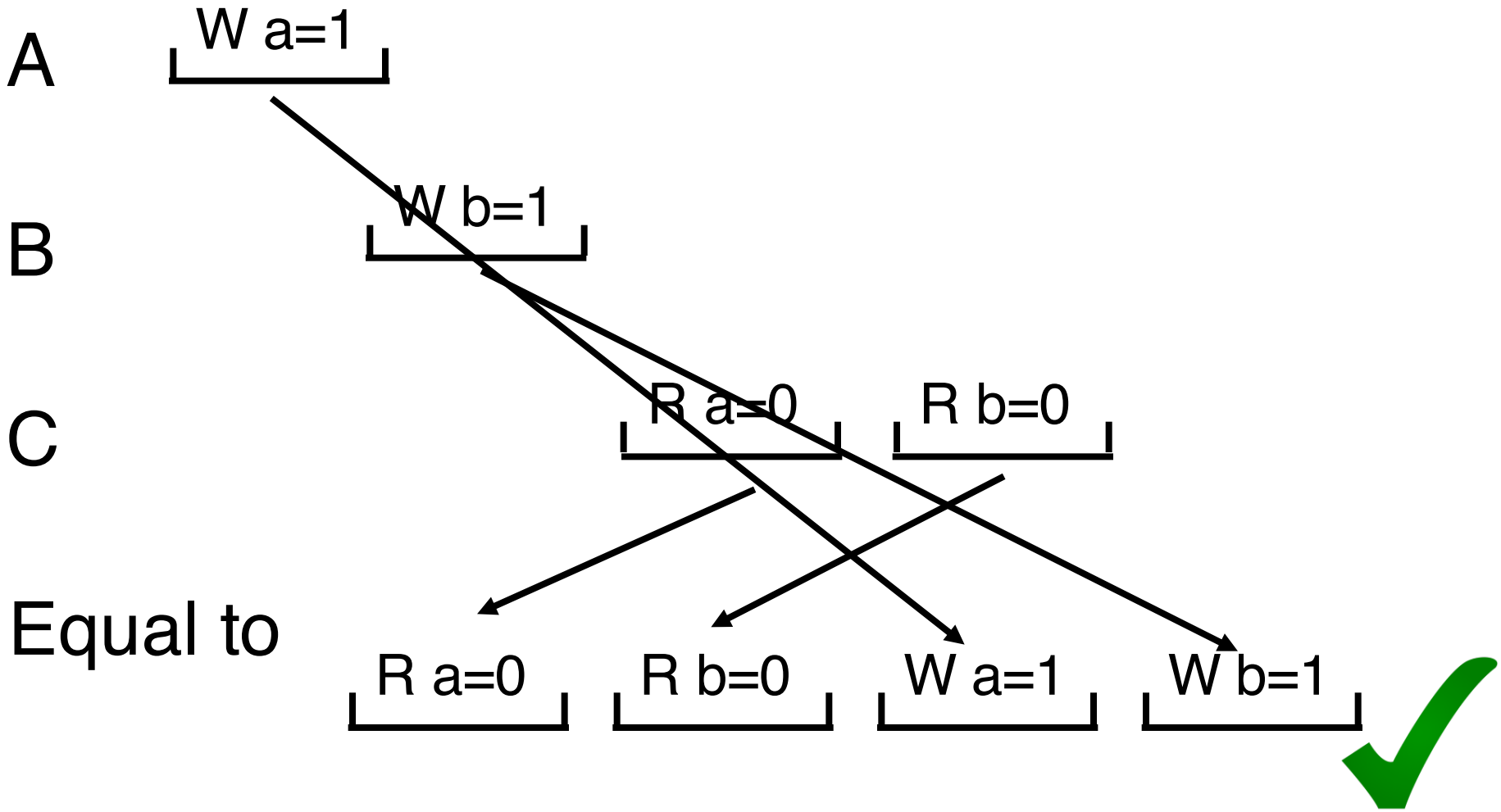
Sequential consistency

- Definition: The result of any execution is the same as if the (read and write) operations by all processes on the data store were executed in some sequential order and the operations of each individual process appear in this sequence in the order specified by its program.
  - There is a global order to execute requests.
  - Every one sees all requests in the same order.

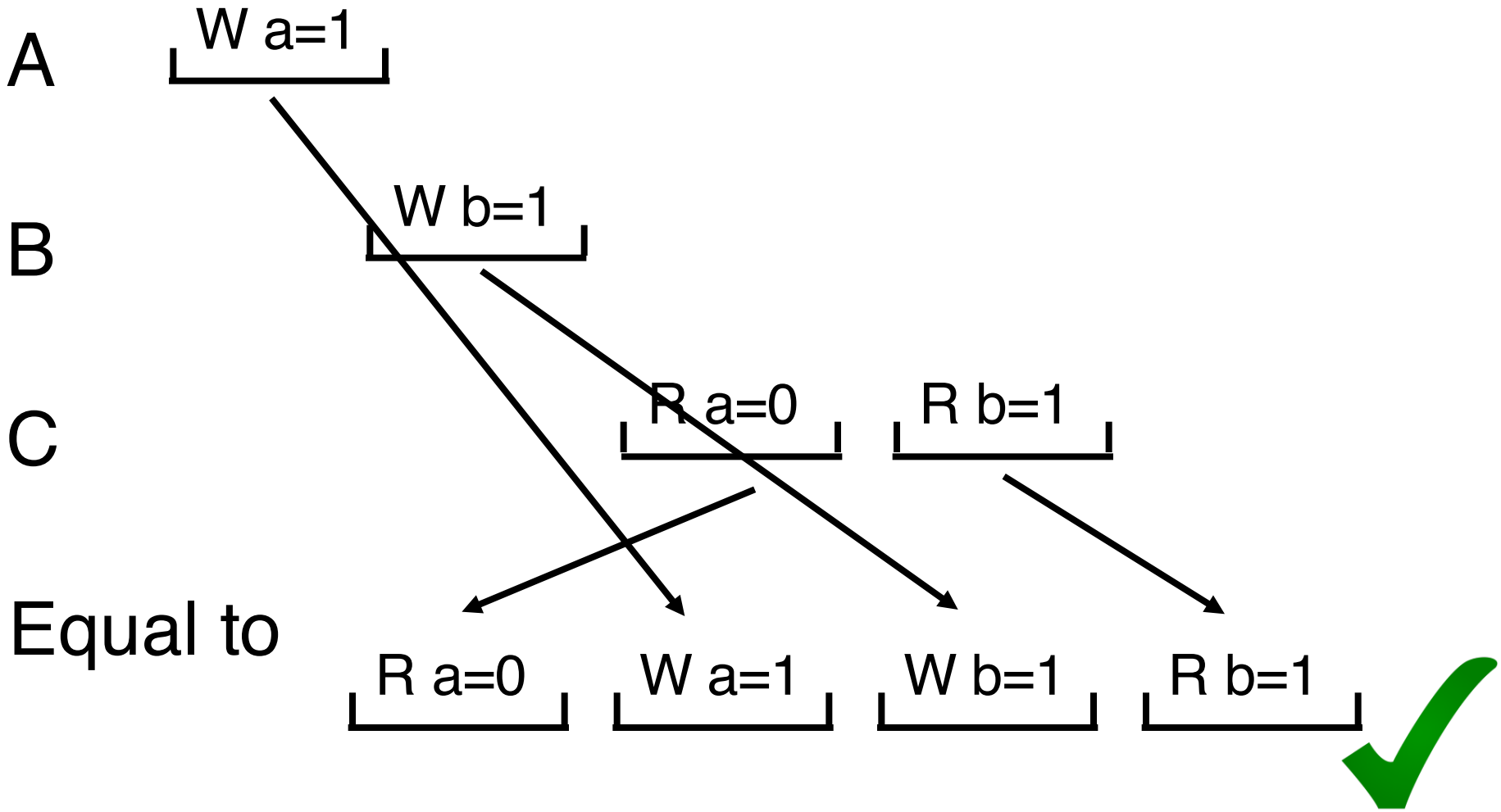
Sequential consistency - Examples



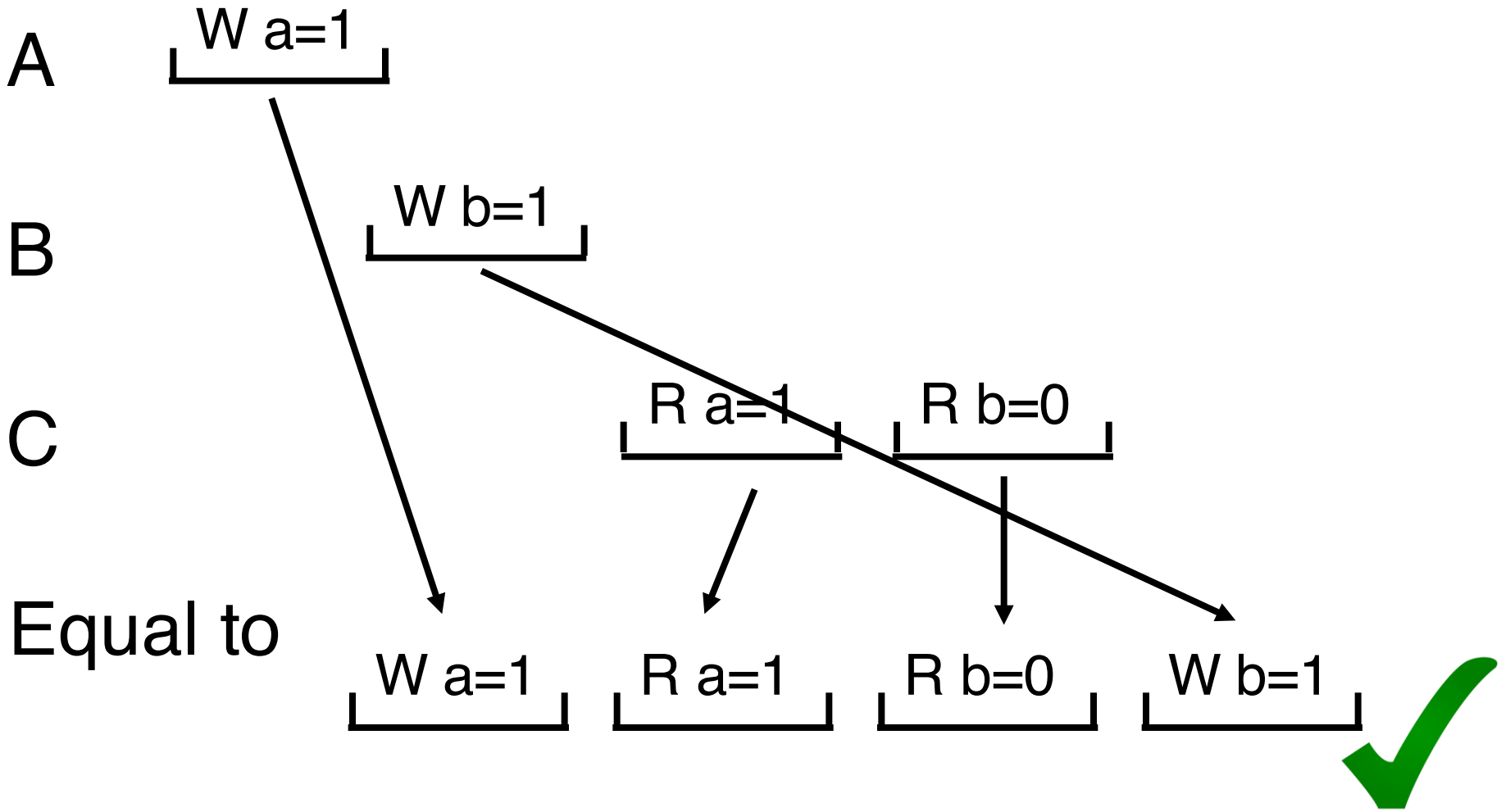
Sequential consistency - Examples



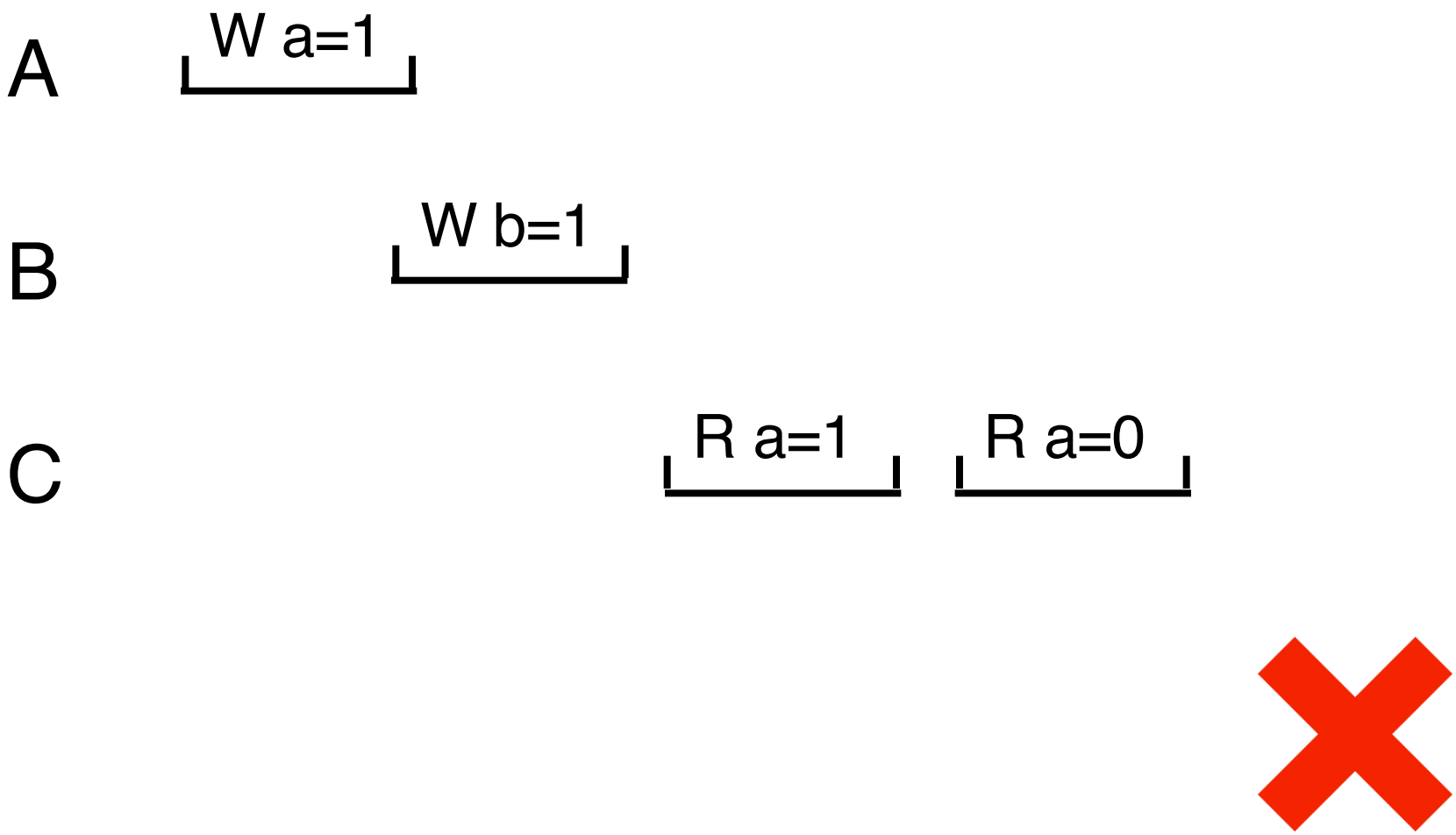
Sequential consistency - Examples



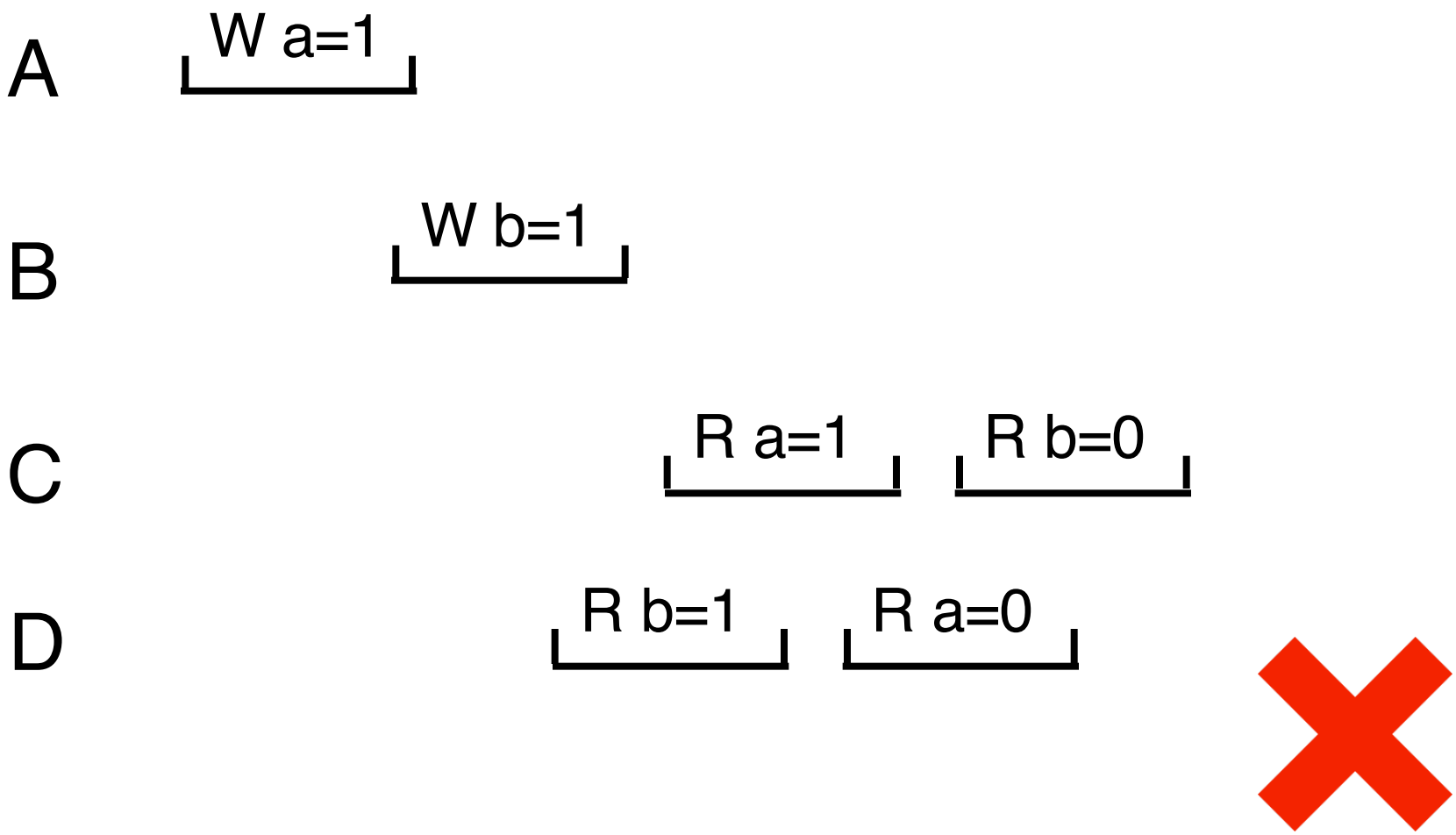
Sequential consistency - Examples



Sequential consistency - Examples



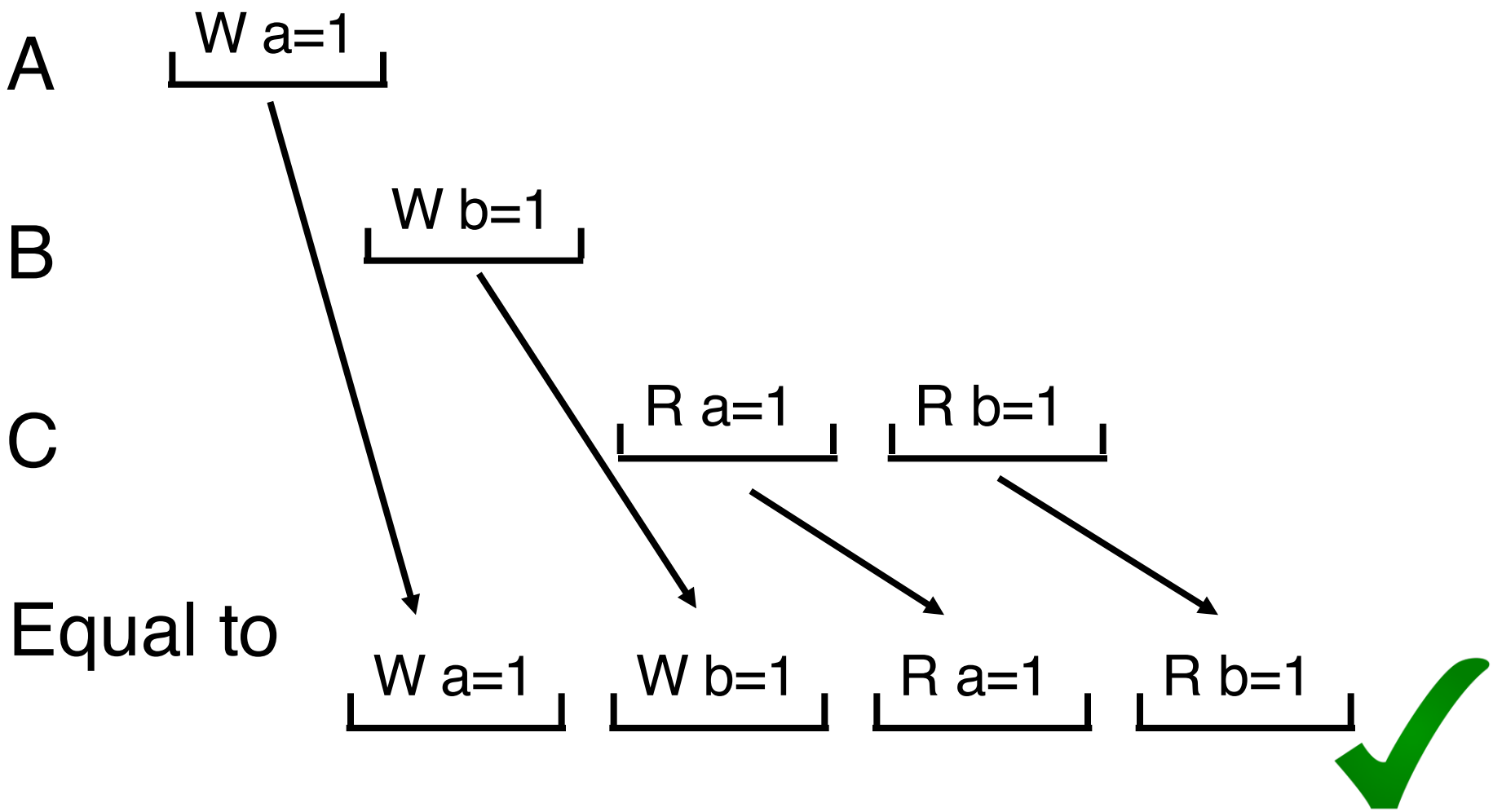
Sequential consistency - Examples



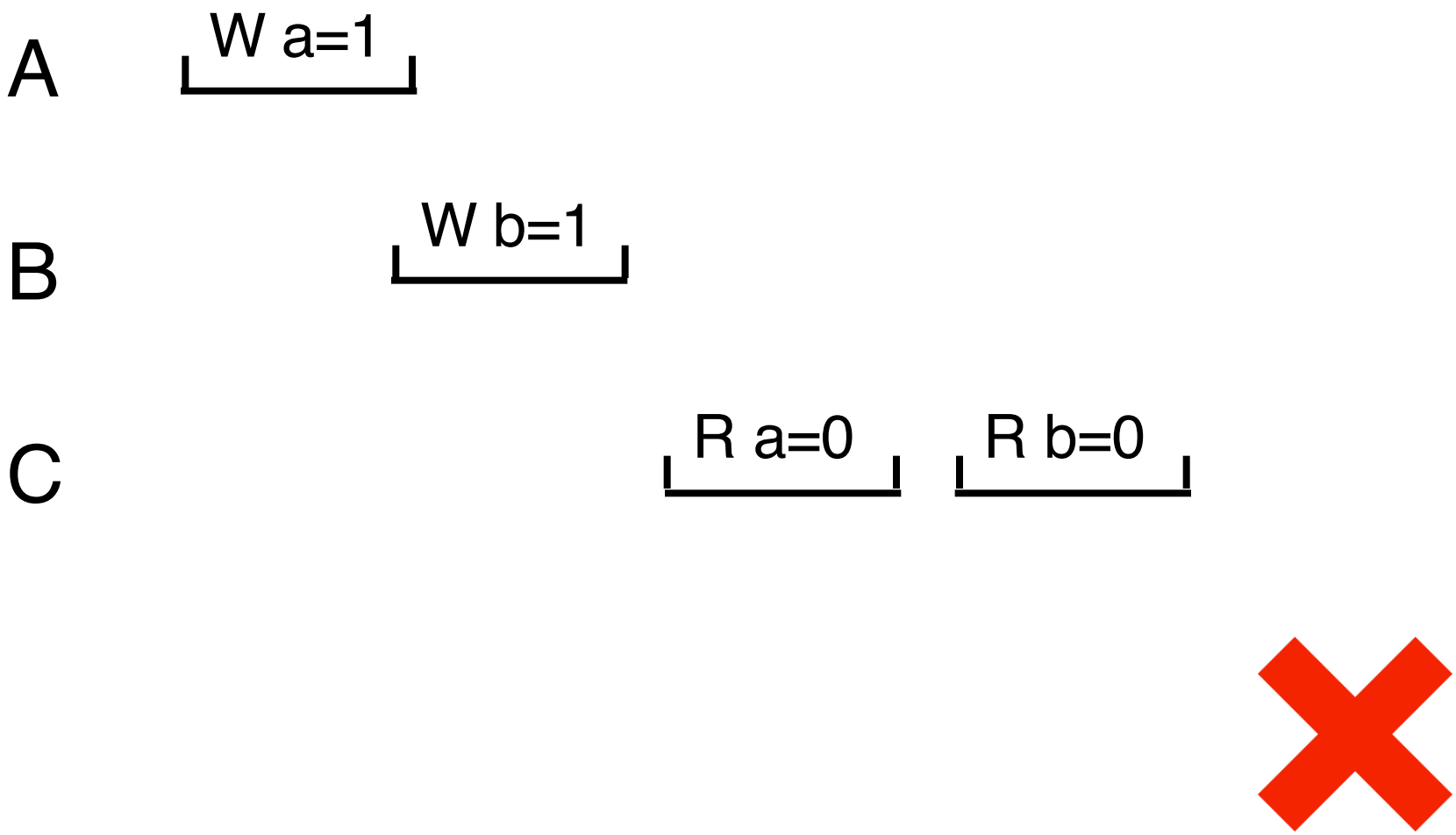
Linearizable

- Definition: sequential consistency + real time constraint
- If a request R1 completes before R2 starts, then R1 must be ordered before R2.
- Concurrent requests can be ordered in either way, but every one must see them in the same order.

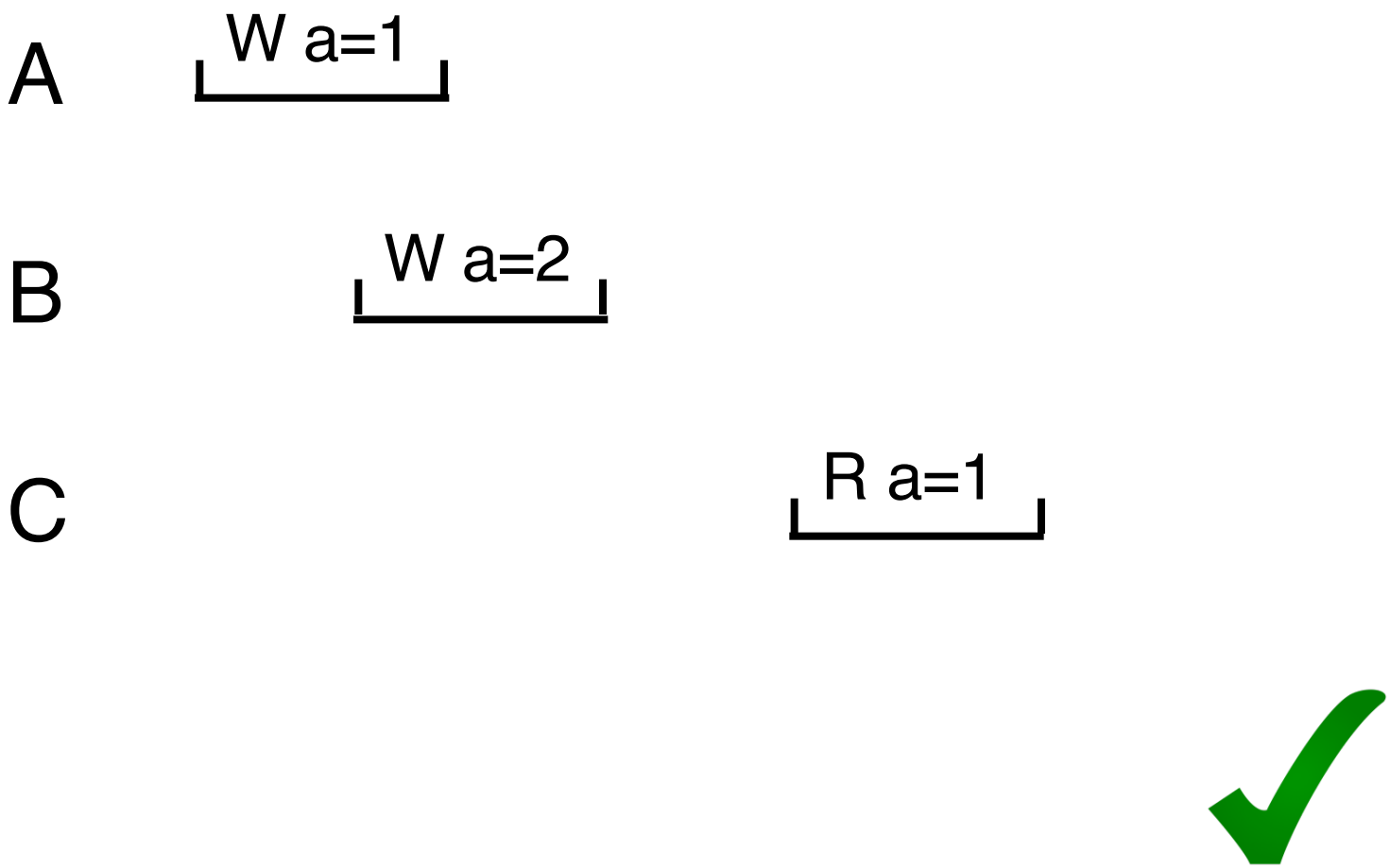
Linearizable - Examples



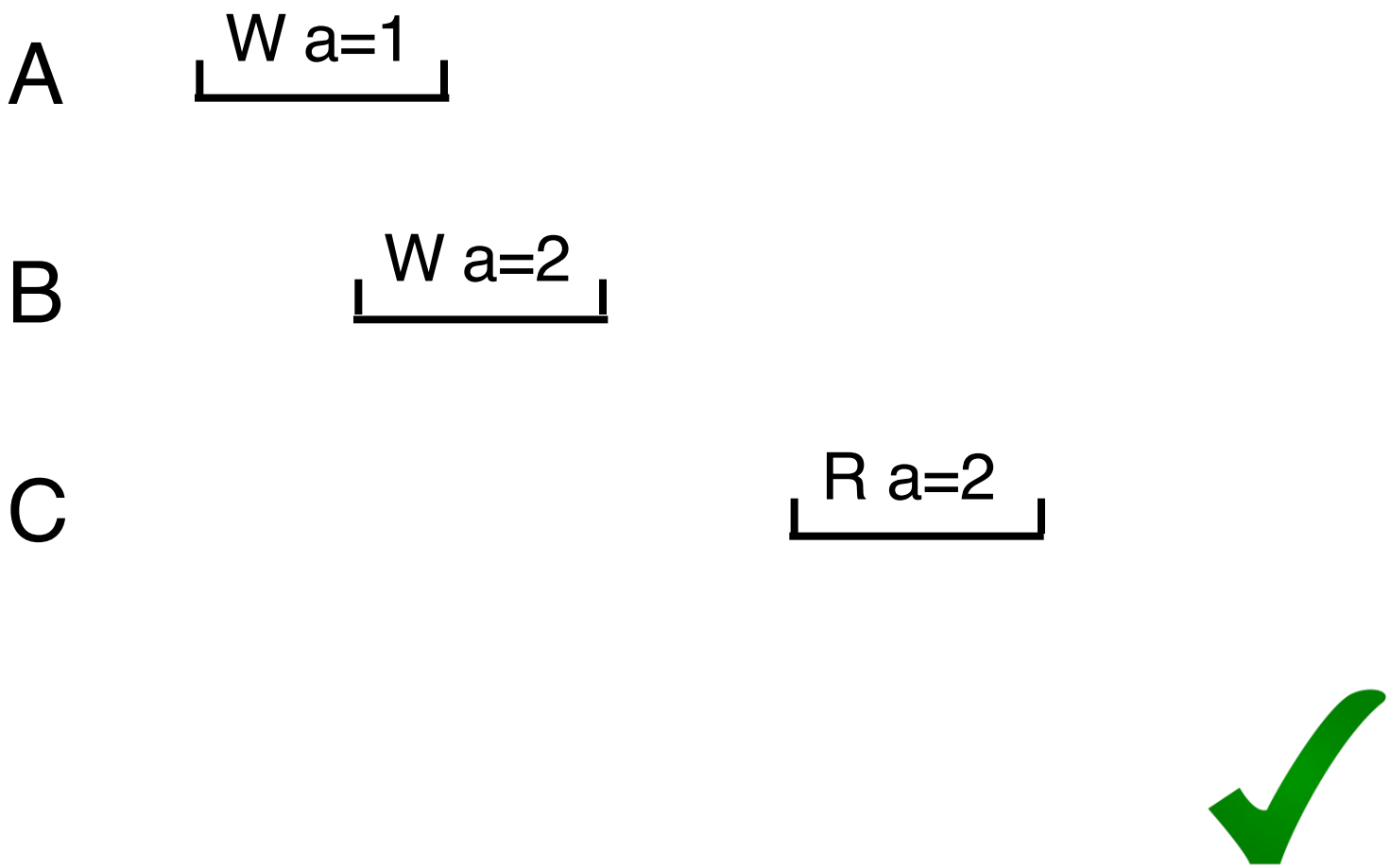
Linearizable - Examples



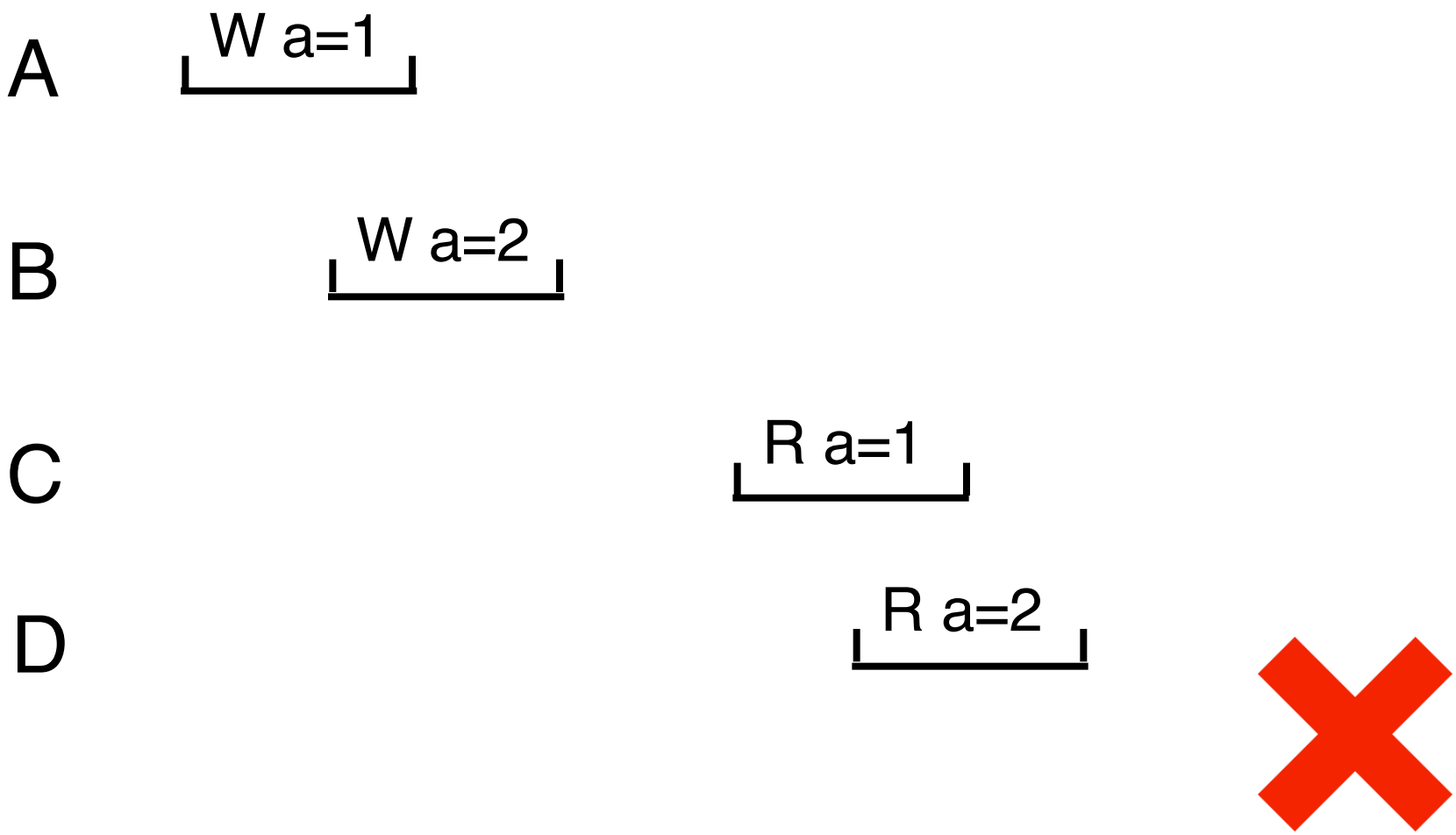
Linearizable - Examples



Linearizable - Examples



Linearizable - Examples



Eventual Consistency

- Definition: If no new updates are made to a given data item, eventually all accesses to that item will return the last updated value
- No global ordering requirement
- Require conflict resolution for concurrent updates

Eventual - Examples

- Version control (e.g. svn, git):
  - Each user makes changes locally
  - A user commits and updates code manually
  - A user may need to resolve conflicts when it updates the repo
  - Different users may see updates in different order

Causal Consistency

- Definition: Respect partial order defined by a user
  - If a user sees a request R, then it must also see all requests R depends on.

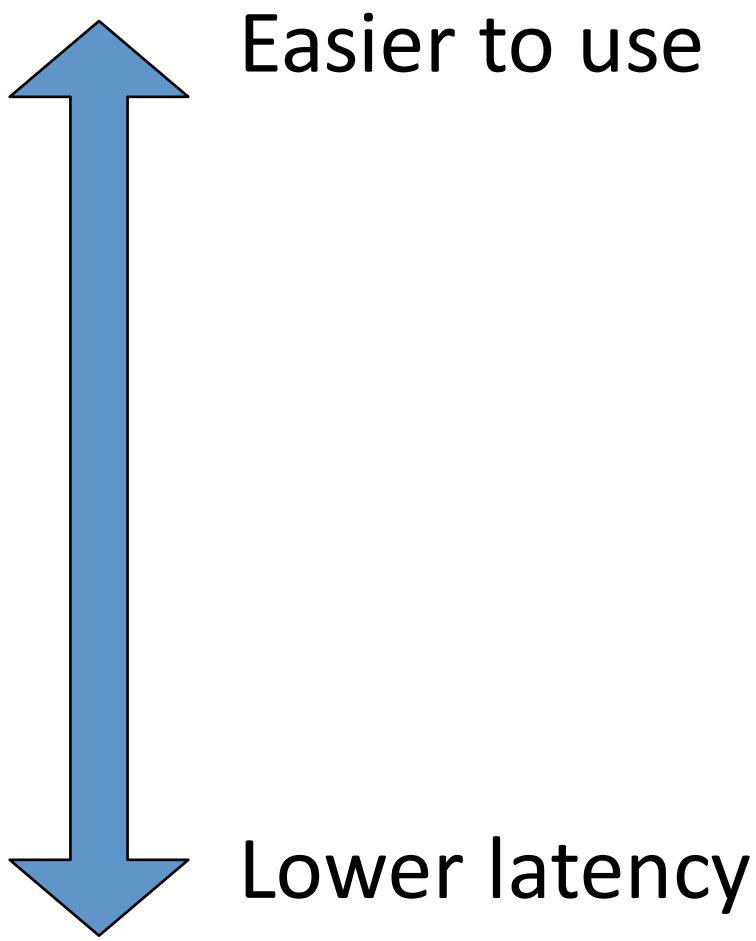
- No global ordering requirement
- Require conflict resolution for concurrent updates

Causal - Examples

- Access control in social network
  - Remove your parents from access list
  - Post a photo that you hope your parents never see

Using different consistency models

- Linearizable
- Sequential
- Causal
- Eventual



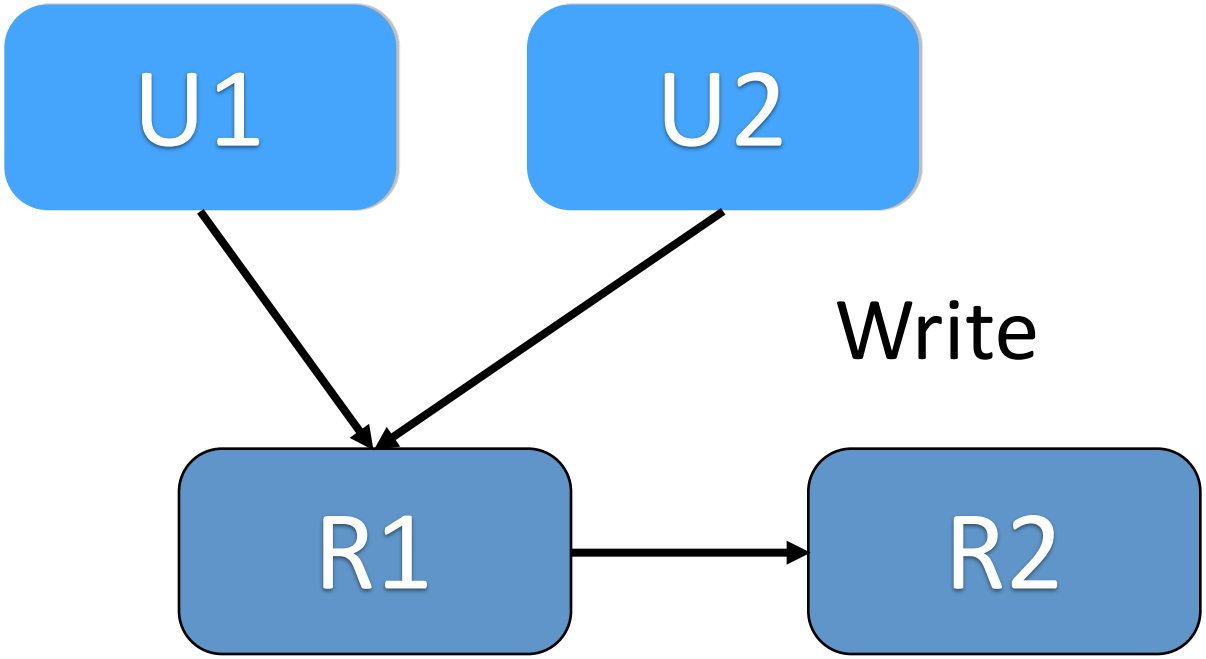
Using different consistency models

- Linearizable: almost don’t need to worry about anything
- Sequential: may read stale data
- Causal: different clients may see events in different orders; need to resolve conflict
- Eventual: almost no guarantee

Can you think about an application that requires global ordering?

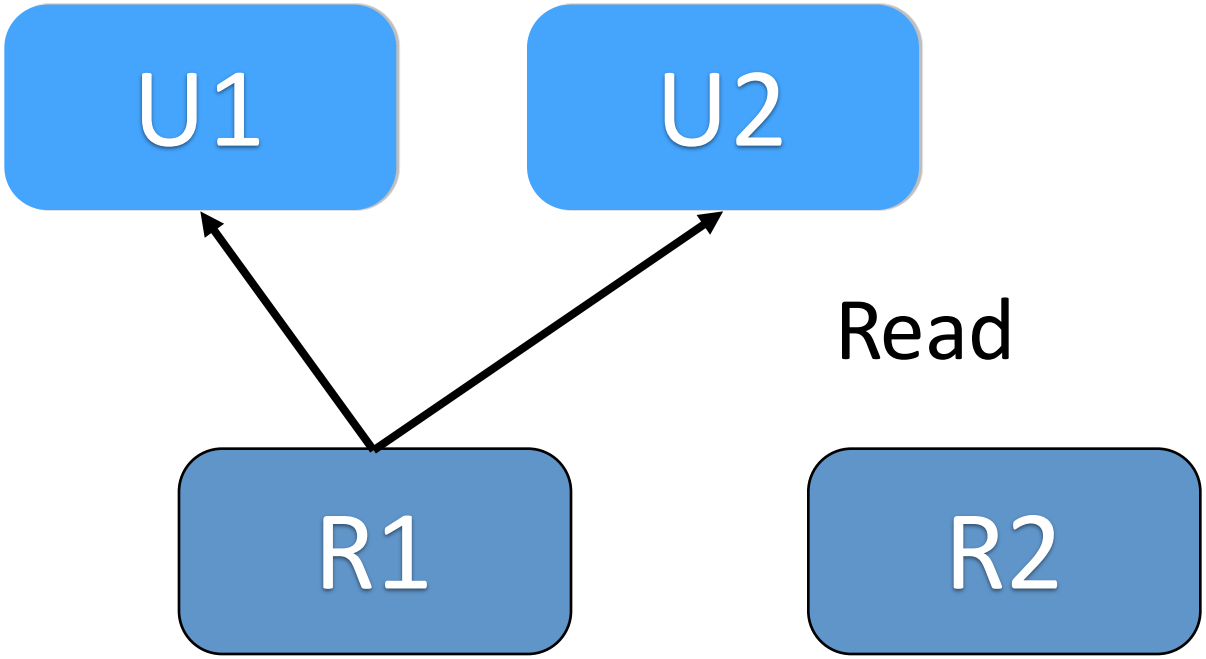
Implementing consistency

Linearizability



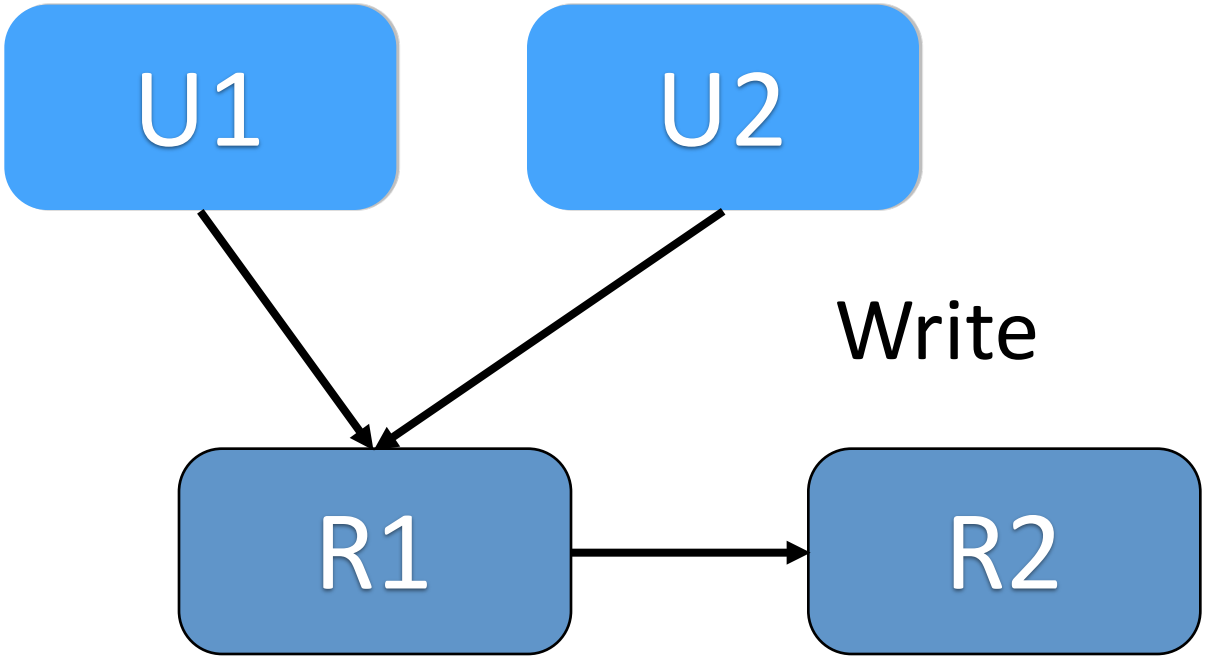
Implementing consistency

Linearizability



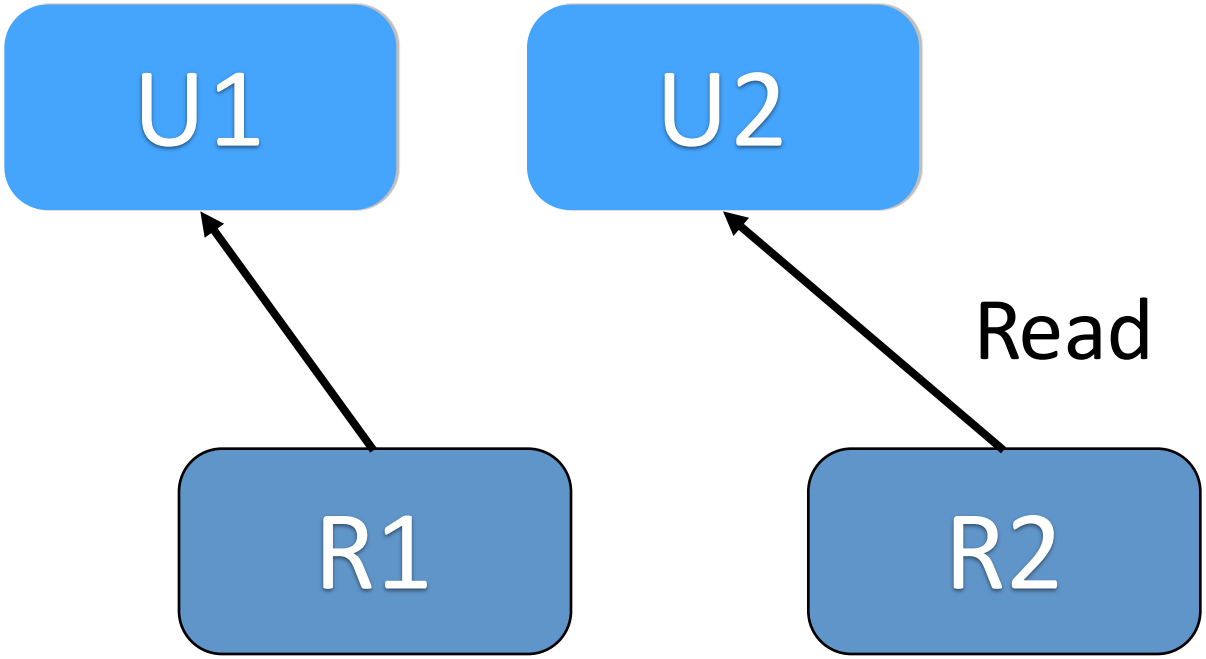
Implementing consistency

Sequential



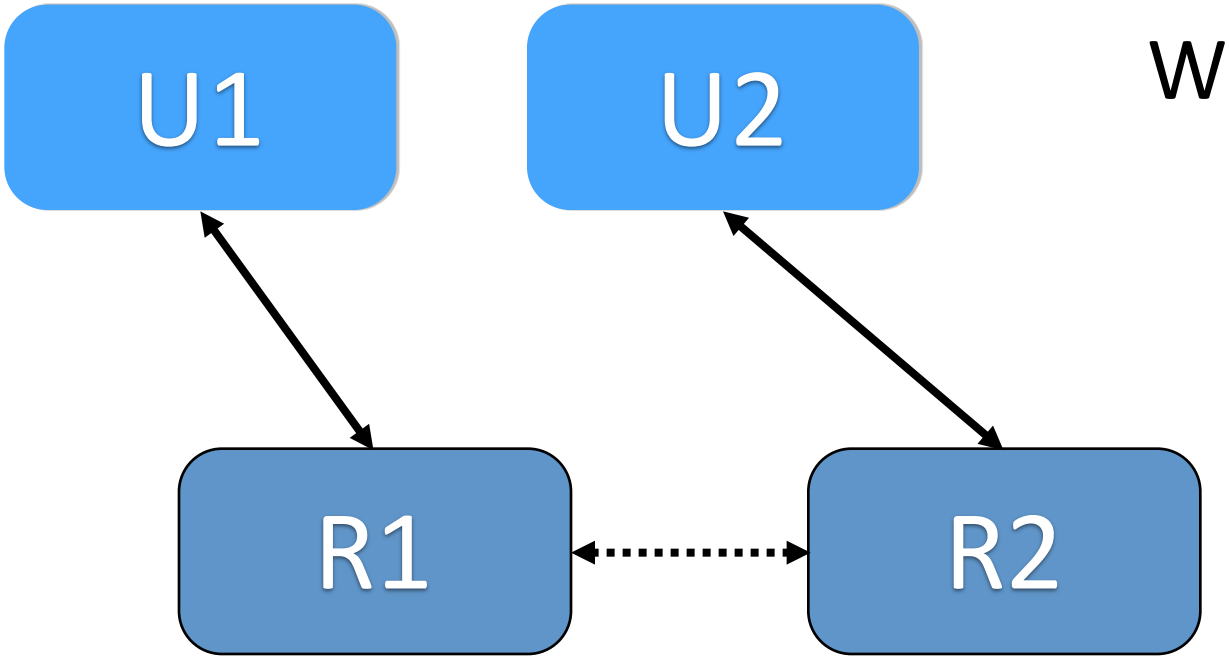
Implementing consistency

Sequential



Implementing consistency

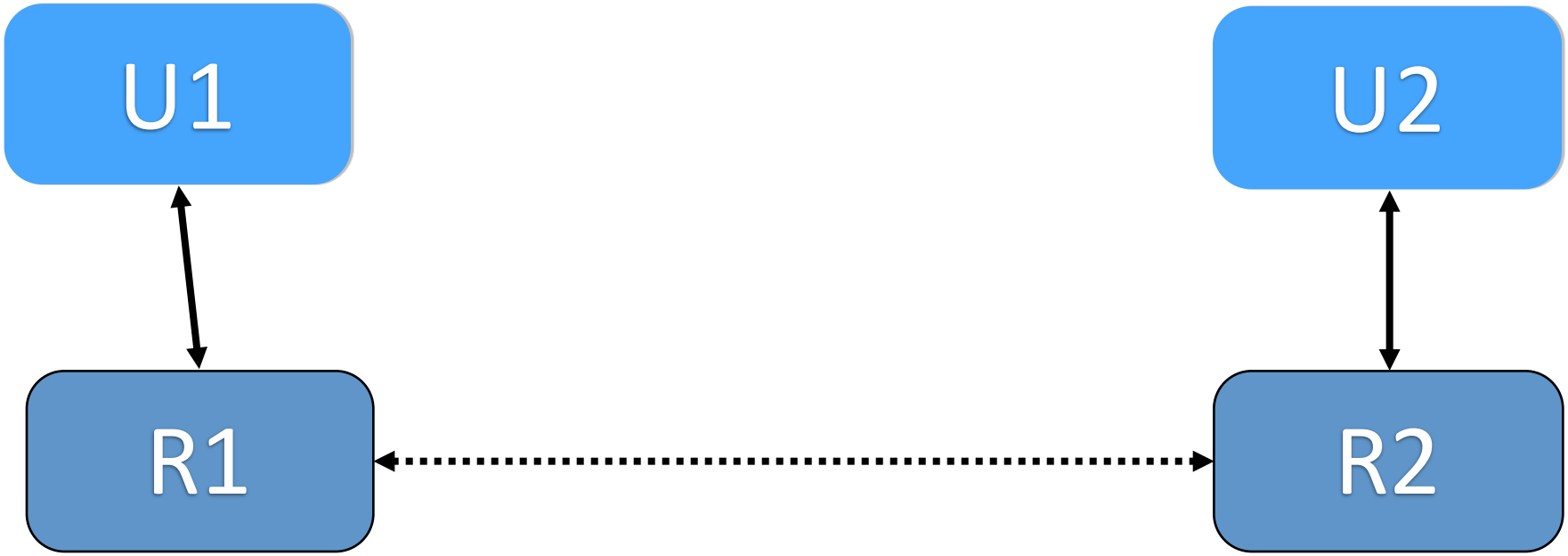
Eventual/Causal



Write/Read

Propagate data in background

Consistency vs Latency

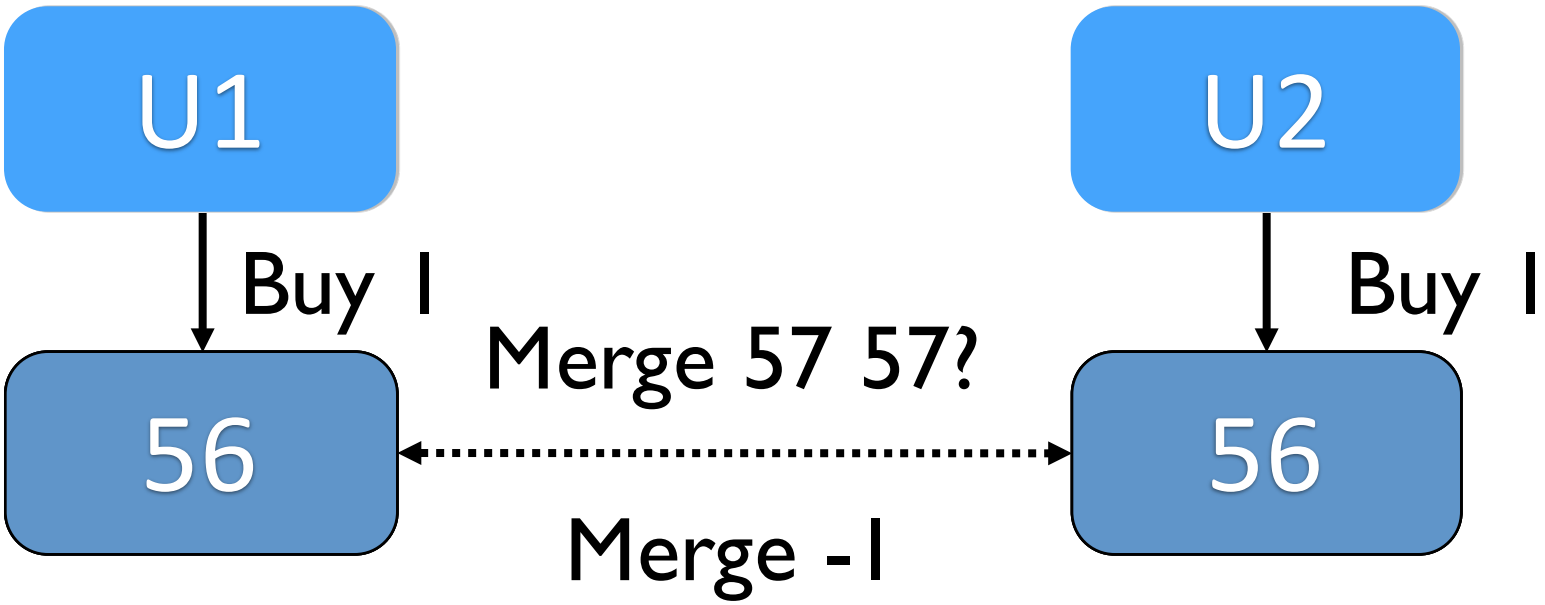


CAPS theorem: Impossible to achieve all the following properties together.

- Sequential consistency (or stronger)
- Availability
- Ability to tolerate network partitions

Challenge: Resolve conflicts

- It is application specific
- Example: Number of remaining tickets



Consistency in reality

- Who are using strong consistency?
  - Online gaming
  - Google

- Who are using weak consistency?
  - Banks (ATMs)
  - Version control
  - Amazon