

Linearizability

- A Quick Overview

Ashvin Goel

Electrical and Computer Engineering
University of Toronto

ECE1724

These slides are modified versions of slides from Michael Freedman & Wyatt Lloyd's course on Distributed Systems

Data Consistency Models

- Contract (or a set of guarantees) that a system provides to applications about expected behavior when data is accessed (read, written, updated, etc.)

Linearizability

- Assumption: Each operation accesses one data item
- All operations are performed in some **total** order
- The total order preserves the **real-time ordering** between operations
 - If operation A **completes** before operation B **begins** in real-time, then A is ordered before B
 - If neither A nor B completes before the other begins, then there is no real-time order, but there must be some total order

Understanding Linearizability

- Writes are ordered
 - Writes appear to occur instantaneously
- Reads read latest data
 - After a write completes, a later read (in real-time order) returns the value of the write (or later write)
 - Once a read returns a value, all later reads return that value or the value of a later write

Linearizable?

$P_A \vdash w(x=1) \dashv$

$P_B \quad \vdash w(x=2) \dashv$

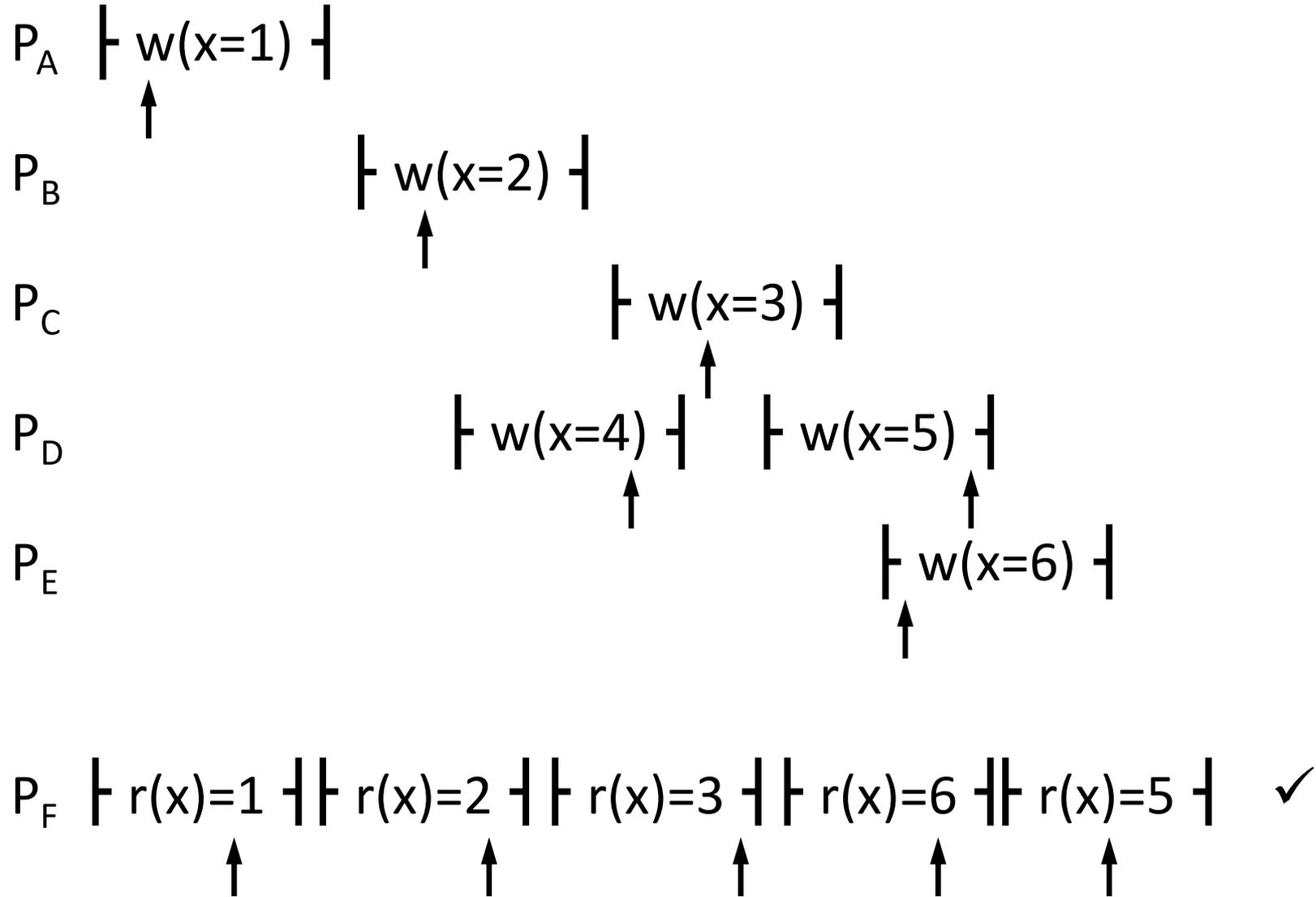
$P_C \quad \quad \quad \vdash w(x=3) \dashv$

$P_D \quad \quad \quad \vdash w(x=4) \dashv \quad \vdash w(x=5) \dashv$

$P_E \quad \quad \quad \quad \quad \quad \vdash w(x=6) \dashv$

$P_F \vdash r(x)=1 \dashv \dashv r(x)=2 \dashv \dashv r(x)=3 \dashv \dashv r(x)=6 \dashv \dashv r(x)=5 \dashv$

Linearizable: Yes



Why Linearizability?

- Behavior is like single machine processing one request at a time
 - Hides complexity associated with concurrency, replication, failures
 - Easier to write correct applications
- Atomic broadcast (Zab protocol used by ZooKeeper), RAFT, PAXOS, etc., provide linearizability for replicated data stores
- However, linearizability is a strong consistency guarantee that can limit performance
 - We will discuss this issue today