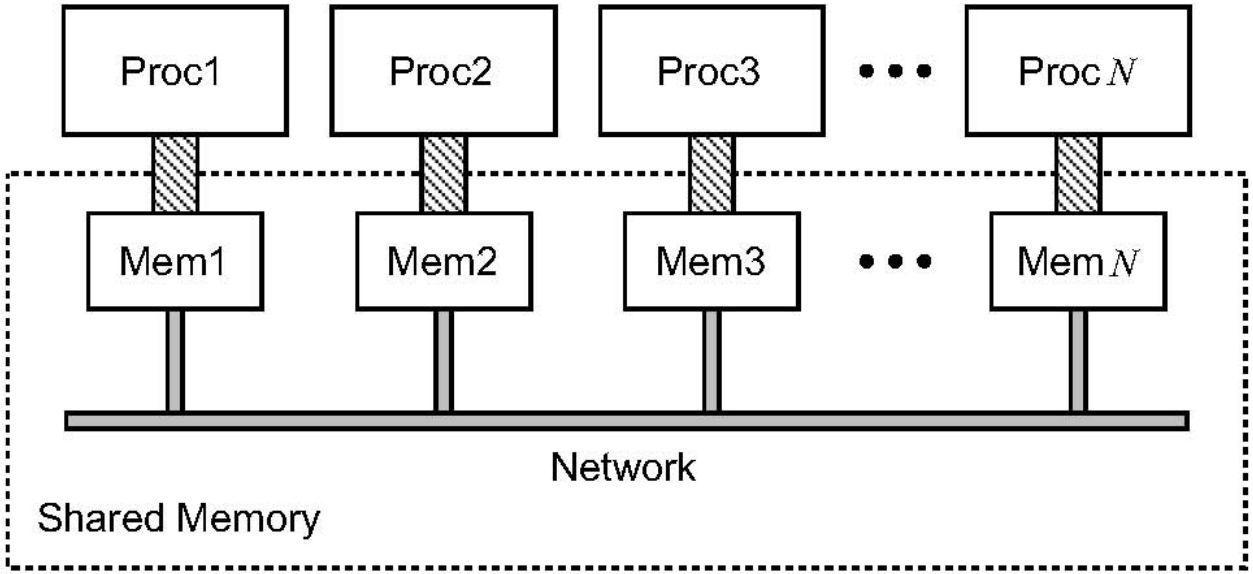
Treadmarks:
Distributed Shared Memory on Standard Workstations and Operating Systems

P. Keleher, A. Cox, S. Dwarkadas, and W. Zwaenepoel
The Winter Usenix Conference 1994

DSM (distributed shared memory)

- A **software system** for parallel computation
  - Shares distributed memories
  - Easier programming
    - Provide a single global address space



DSM (distributed shared memory)

- No widely available DSM implementations
  - In-house research platforms
  - Kernel modifications
  - Poor performance
    - Imitating consistency protocols of hardware
    - False sharing

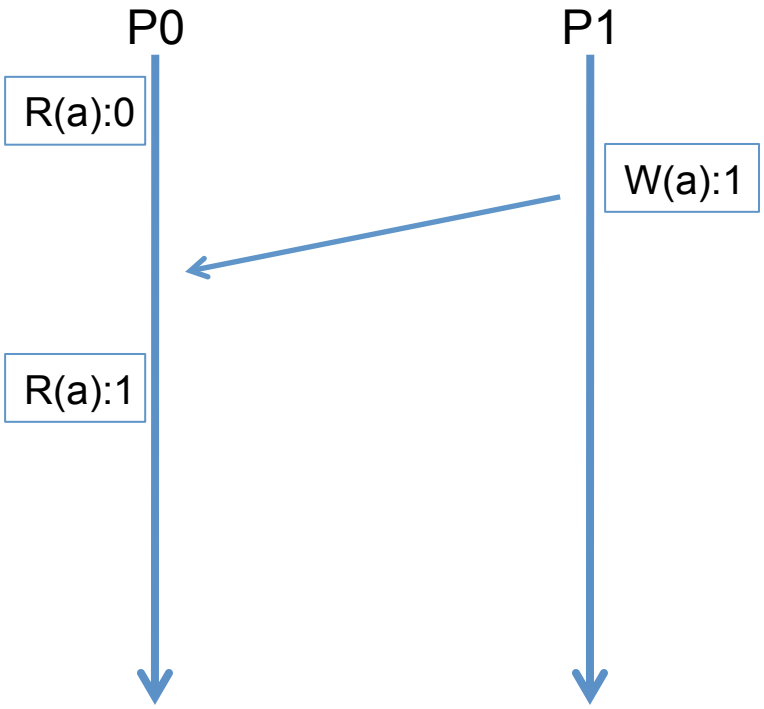
Treadmarks

- **Objectives**
  - Commercially available workstations and OS
    - Standard Unix system on DECstation
  - Efficient user-level DSM implementation
    - Reduce communication overhead

- **Design**
  - LRC (lazy release consistency)
  - Multiple writer protocol
  - Lazy diff creation

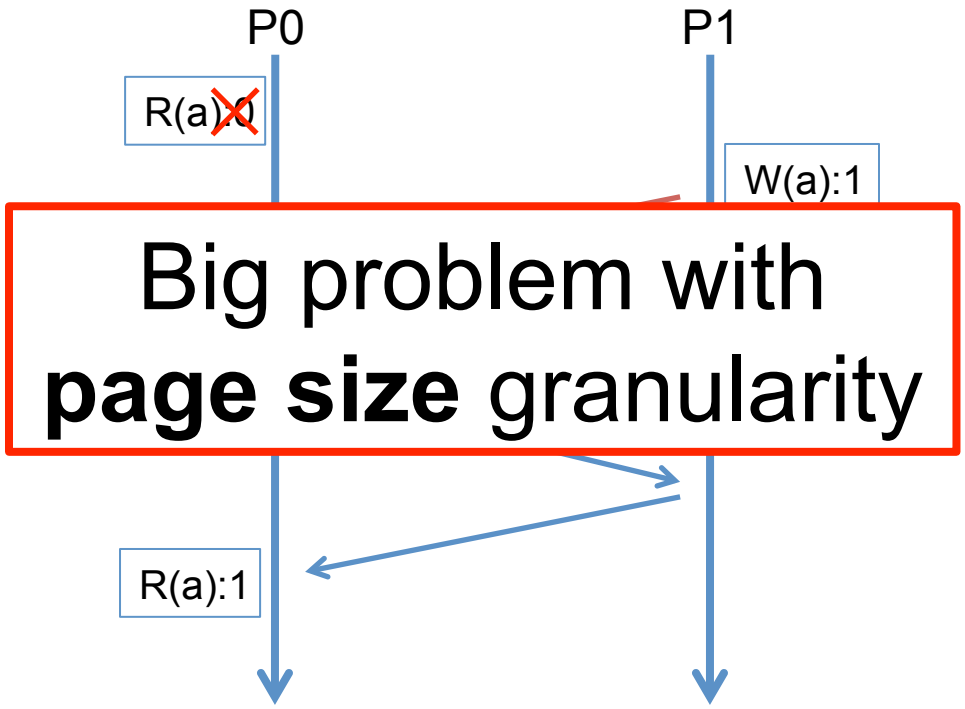
Consistency protocol (SC)

- Sequential Consistency
  - Every write visible “immediately”
  - Single writer



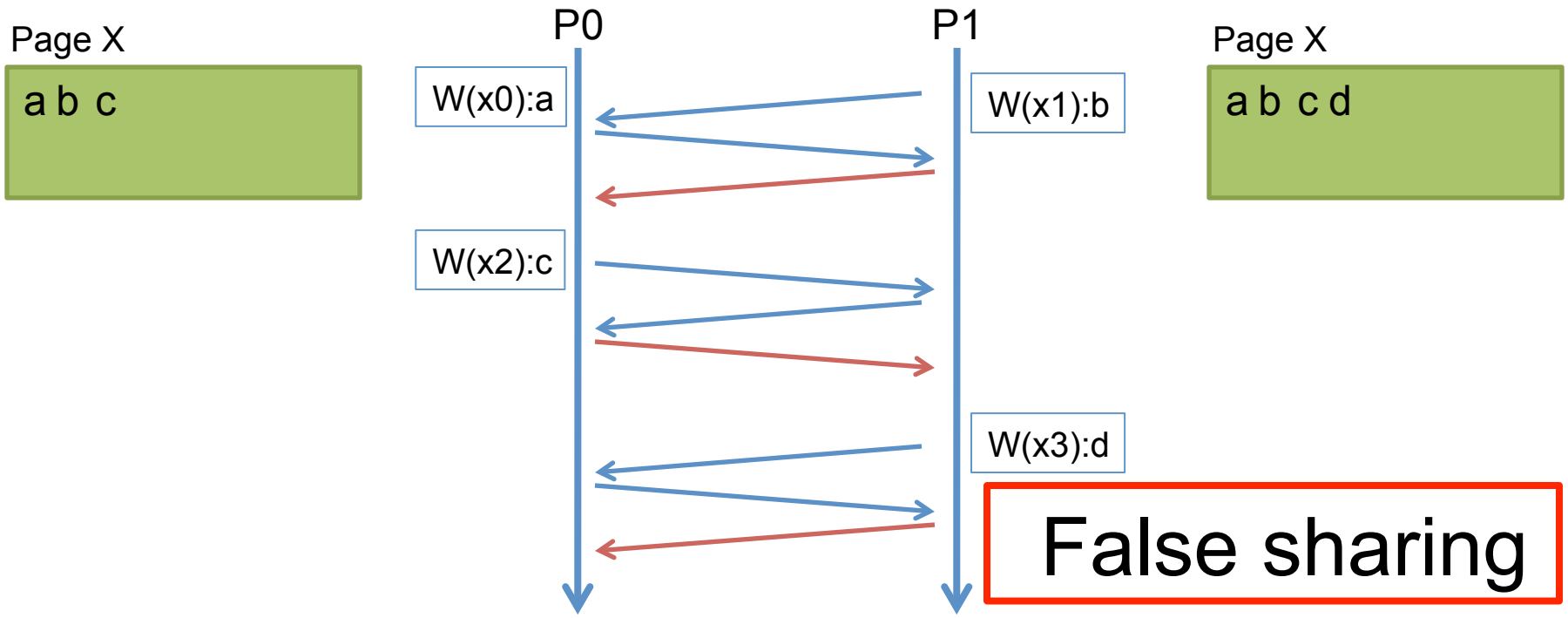
Consistency protocol (SC)

- Sequential Consistency
  - Every write visible “immediately”
  - Single writer



Consistency protocol (SC)

- Sequential Consistency
  - Every write visible “immediately”
  - Single writer



Consistency protocol (RC)

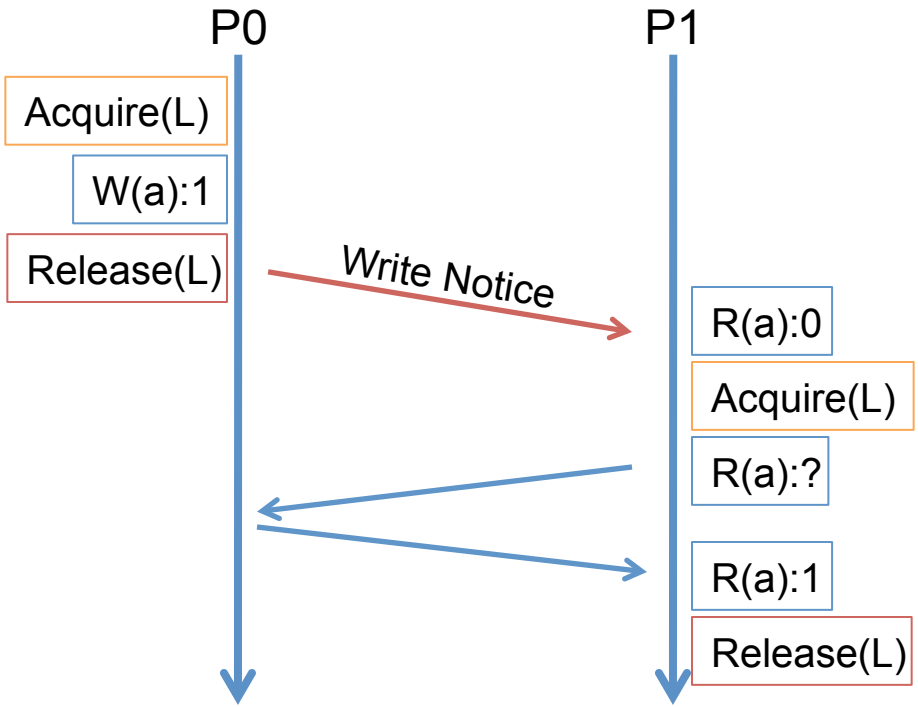
- Release Consistency
  - Relaxed memory consistency model
    - delay making its changes visible to other processors until certain synchronization accesses occurs
  - Synchronization points
    - Acquire(), Release() (similar to locks, barriers)
  - Two types
    - ERC (eager), LRC (lazy)

Consistency protocol (RC)

- Release Consistency
  - Acquire() and release() are sequentially consi stent
    - Release() is performed after all previous operations have completed
    - Operations are performed after previous acquire() have been performed
  - Acquire() and release() pair between conflictin g accesses
    - SC and RC produce the same results.

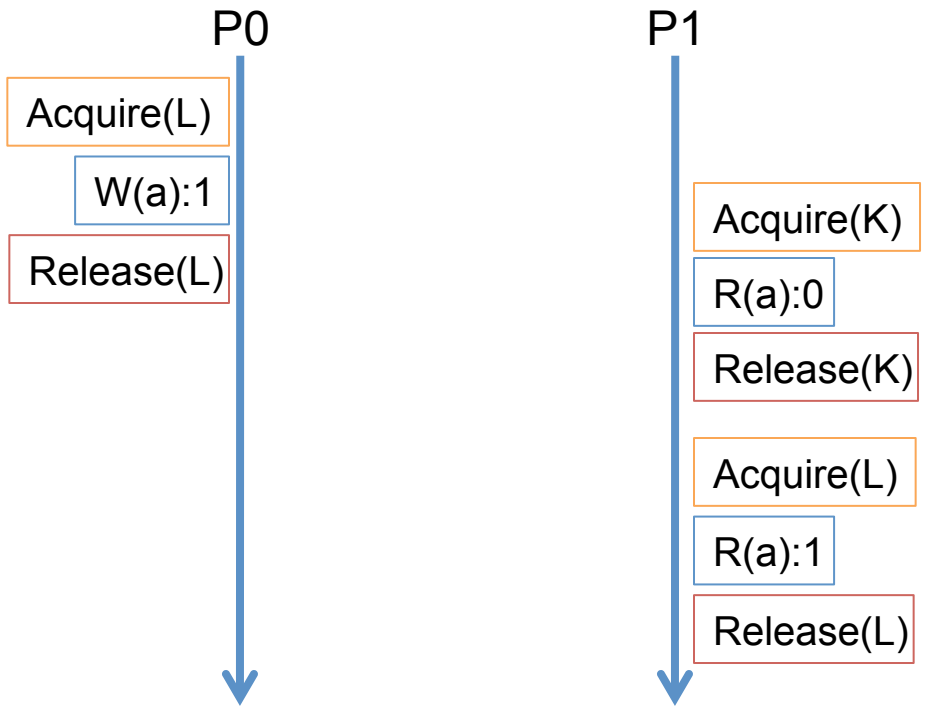
Consistency protocol (RC)

- ERC
  - Write information is delivered at the release p oint



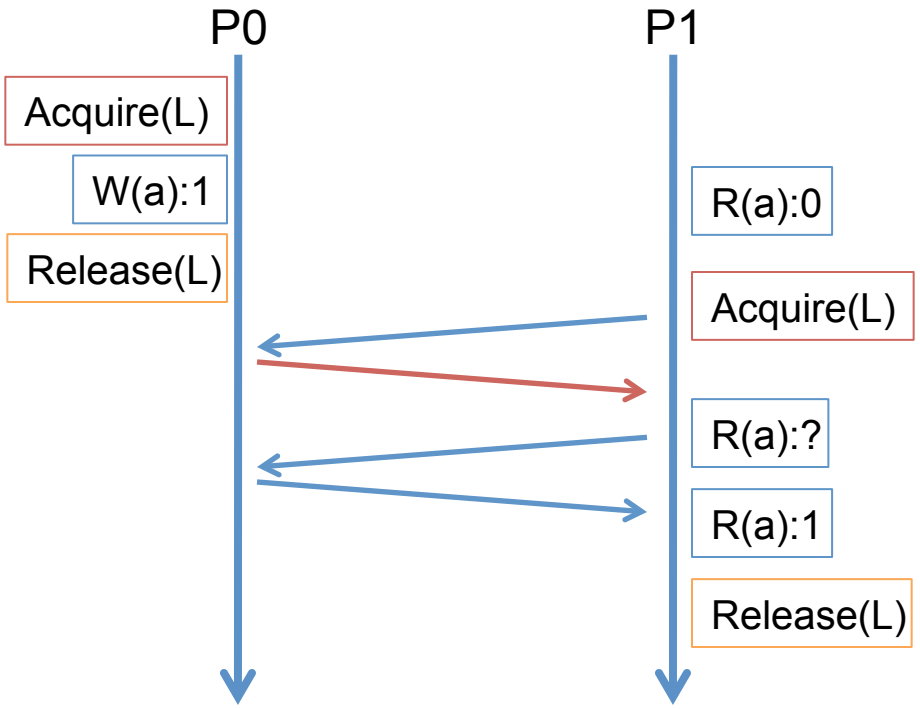
Consistency protocol (RC)

- ERC
  - Write information is delivered at the release p oint



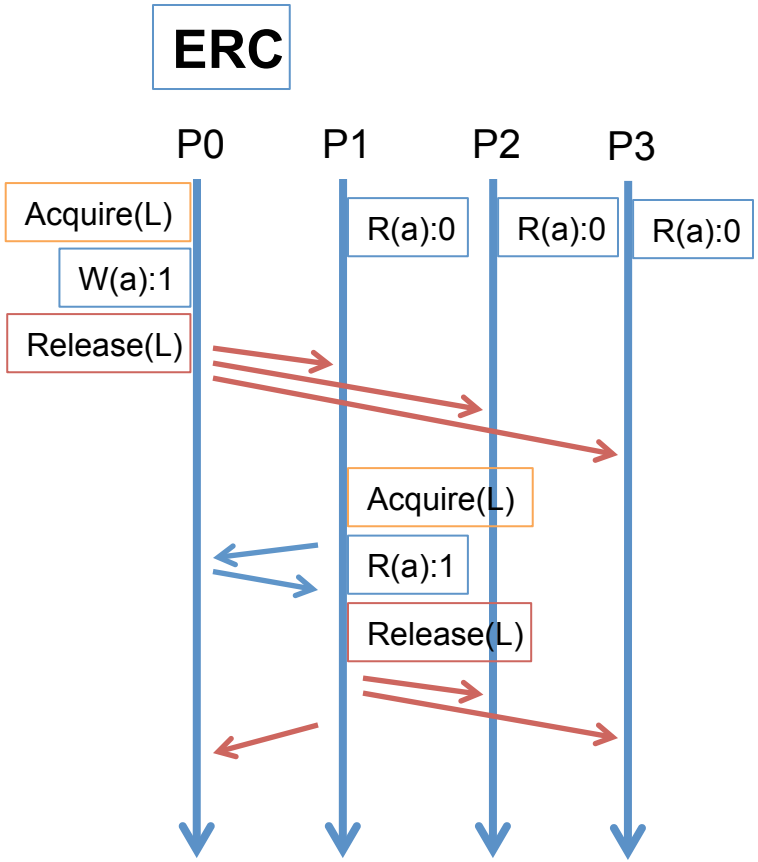
Consistency protocol (RC)

- LRC
  - The delivery is postponed until the acquire
  - Fewer messages than ERC



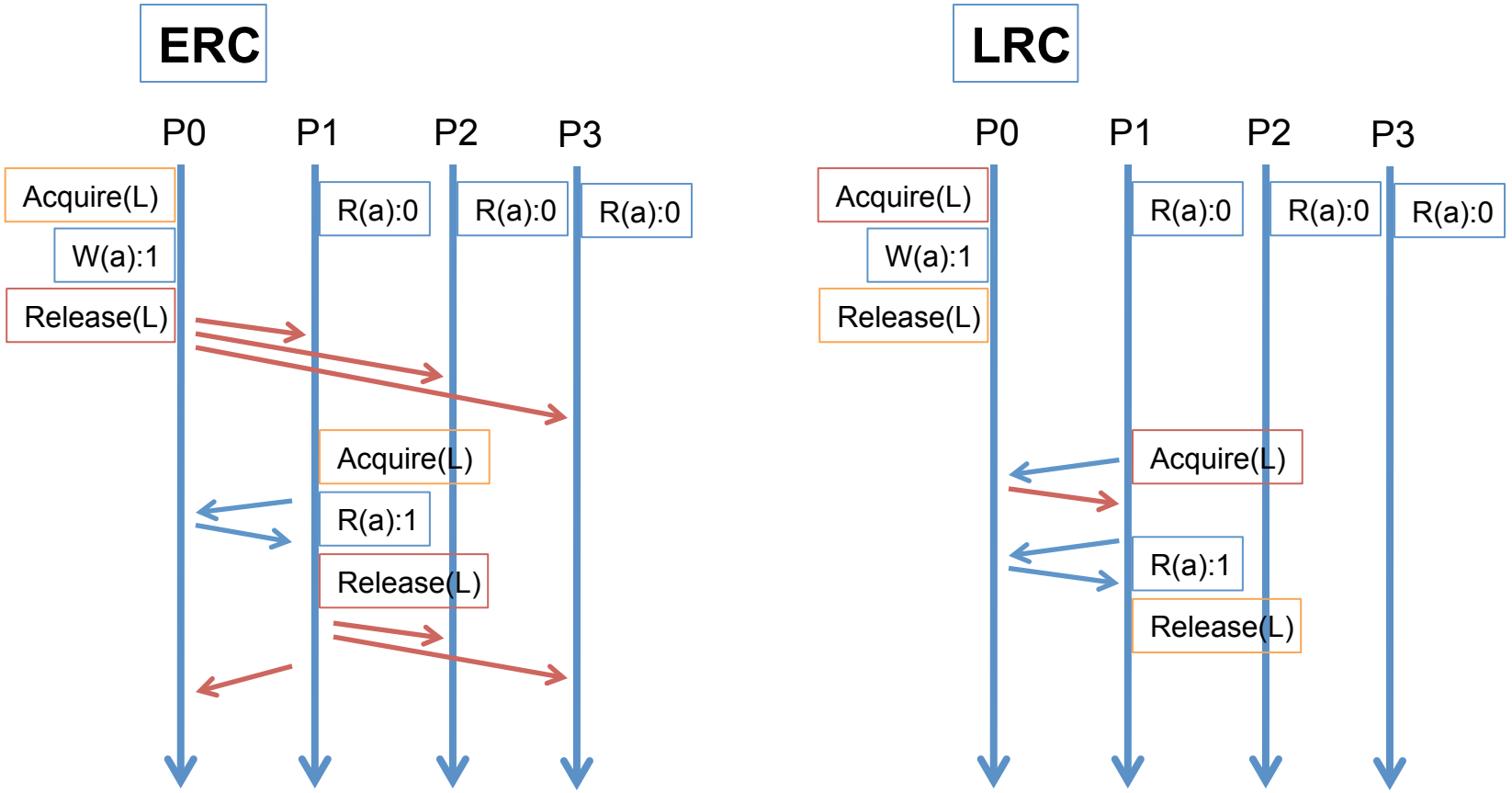
Consistency protocol (RC)

- ERC vs. LRC

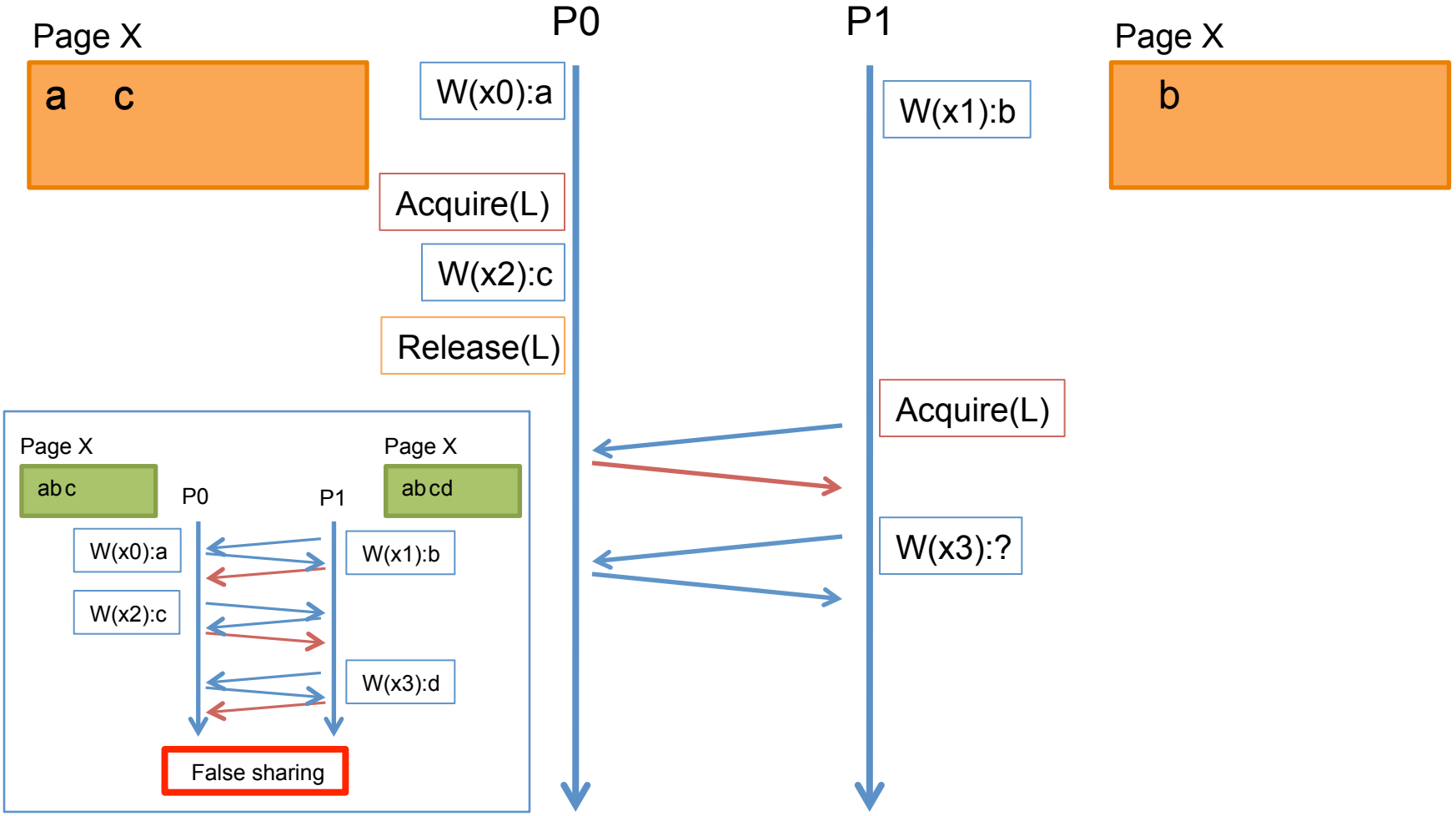


Consistency protocol (RC)

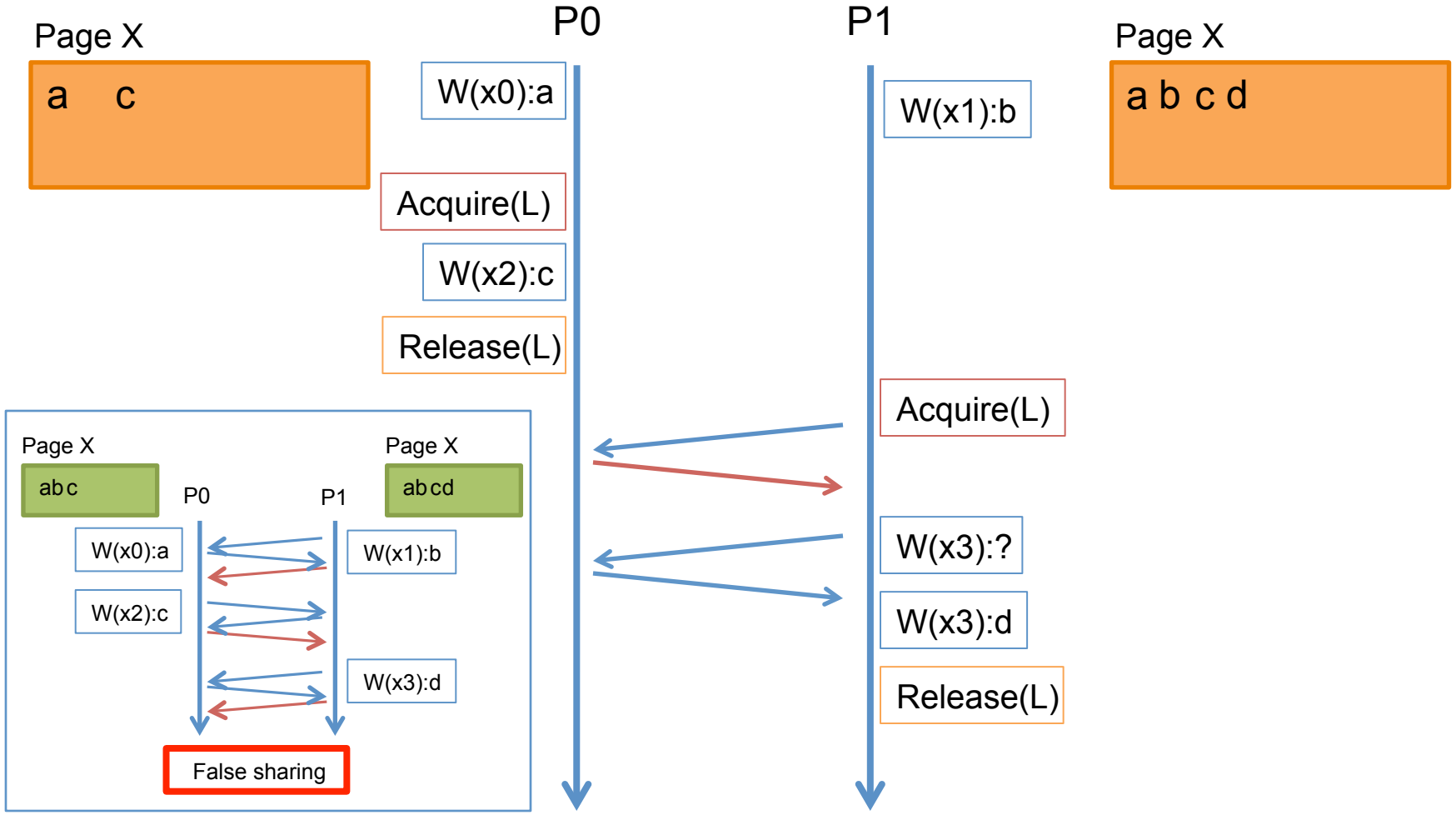
- ERC vs. LRC



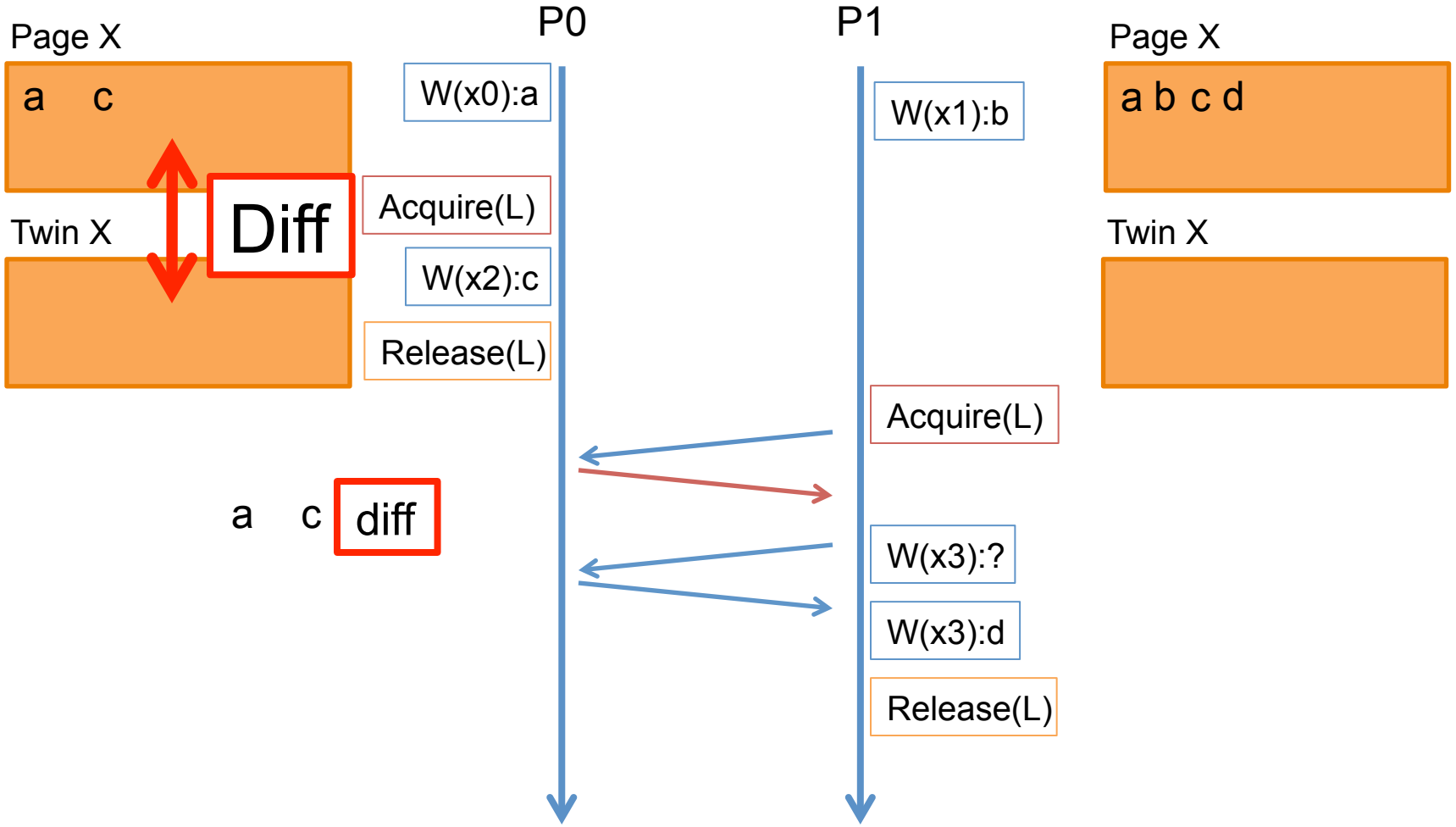
Multiple writer protocol



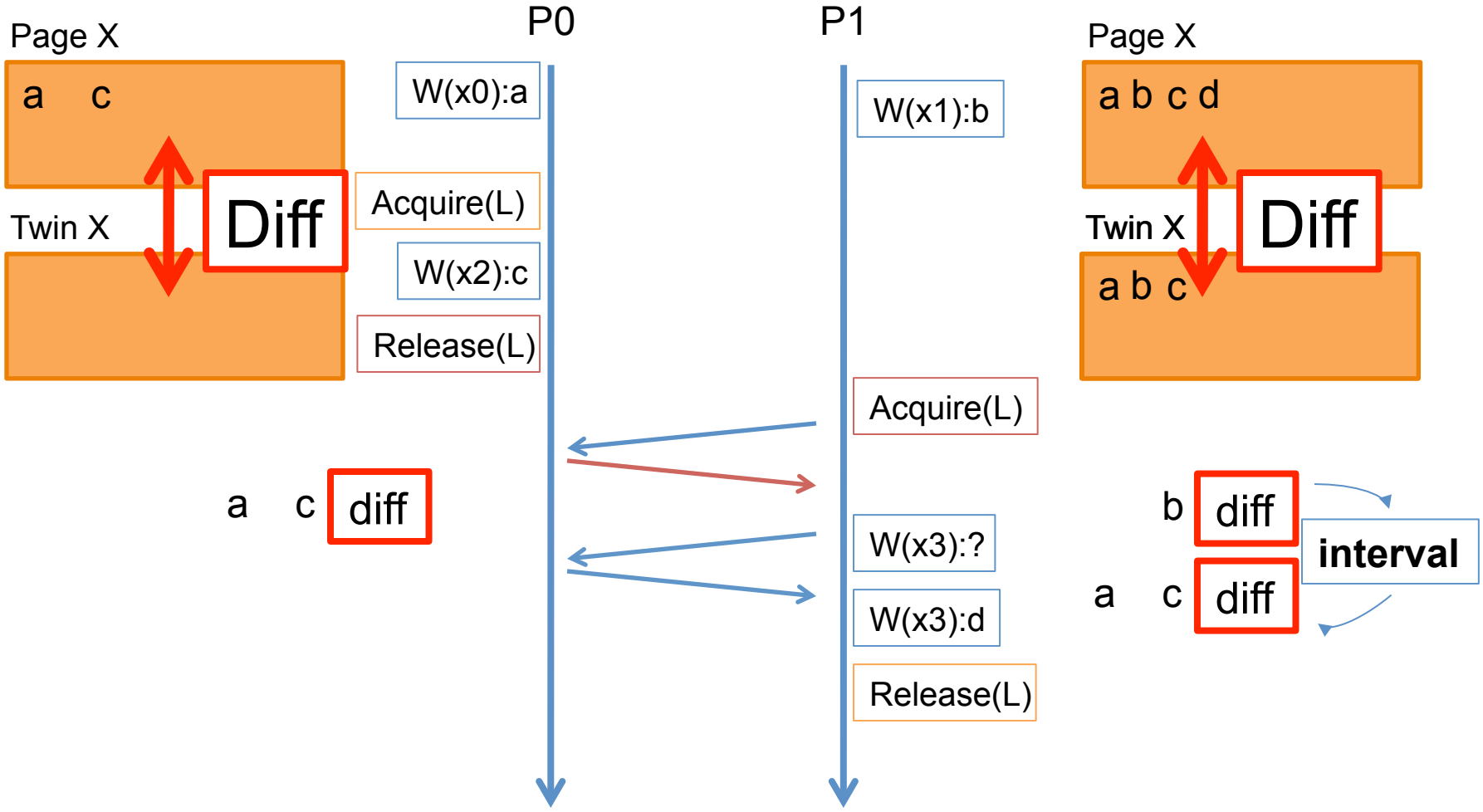
Multiple writer protocol



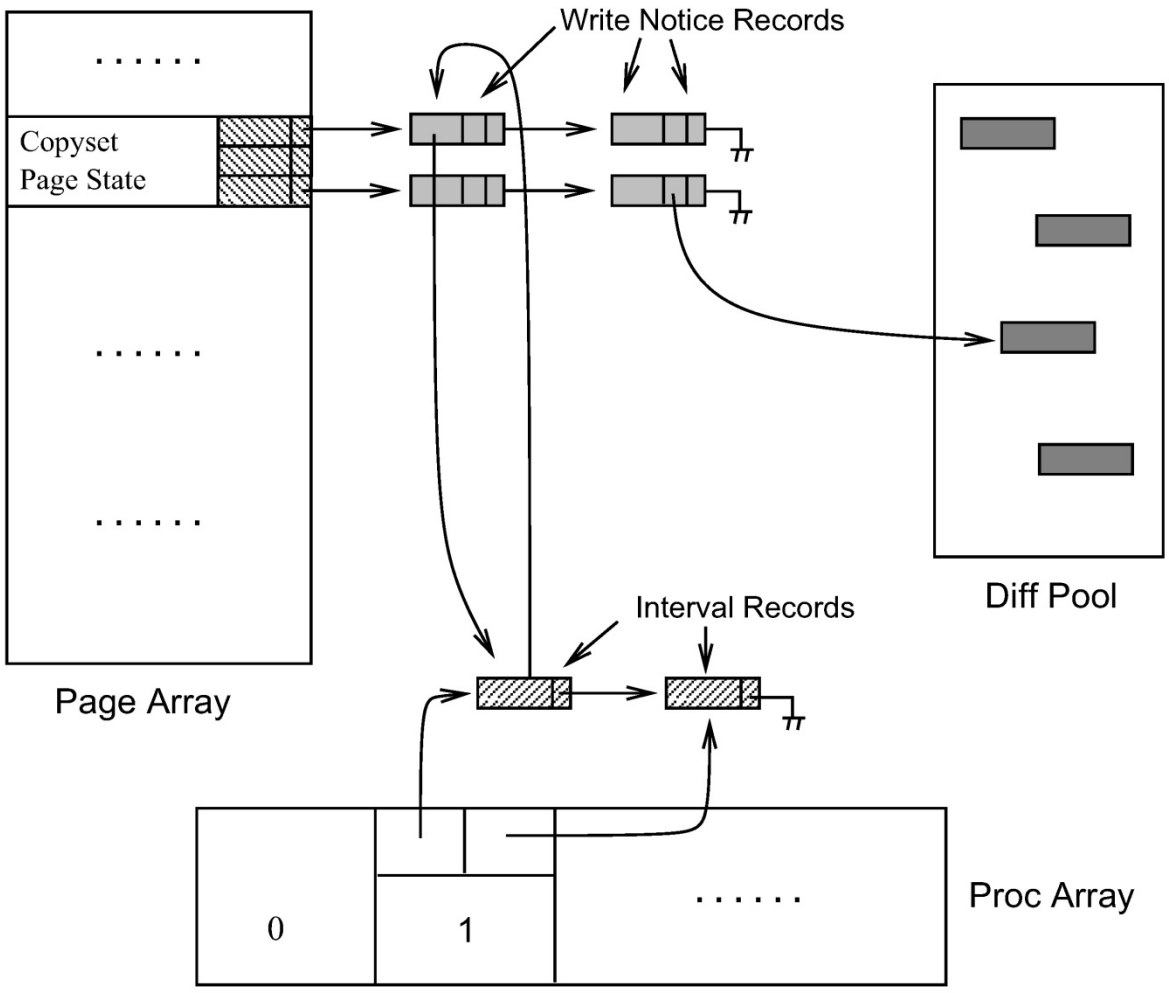
Twin and Diff



Twin and Diff



Implementation



Etc.

- Lock & barrier
  - Statically assigned manager

- Garbage collection
  - reclaim the space used by write notice record s, interval records, and diffs
  - Triggered when the free space drops below a threshold

Evaluation

- Experimental Environment
  - 8 DECstation-5000/240
  - connected to a 100-Mbps ATM LAN and a 10-Mbps Et hernet

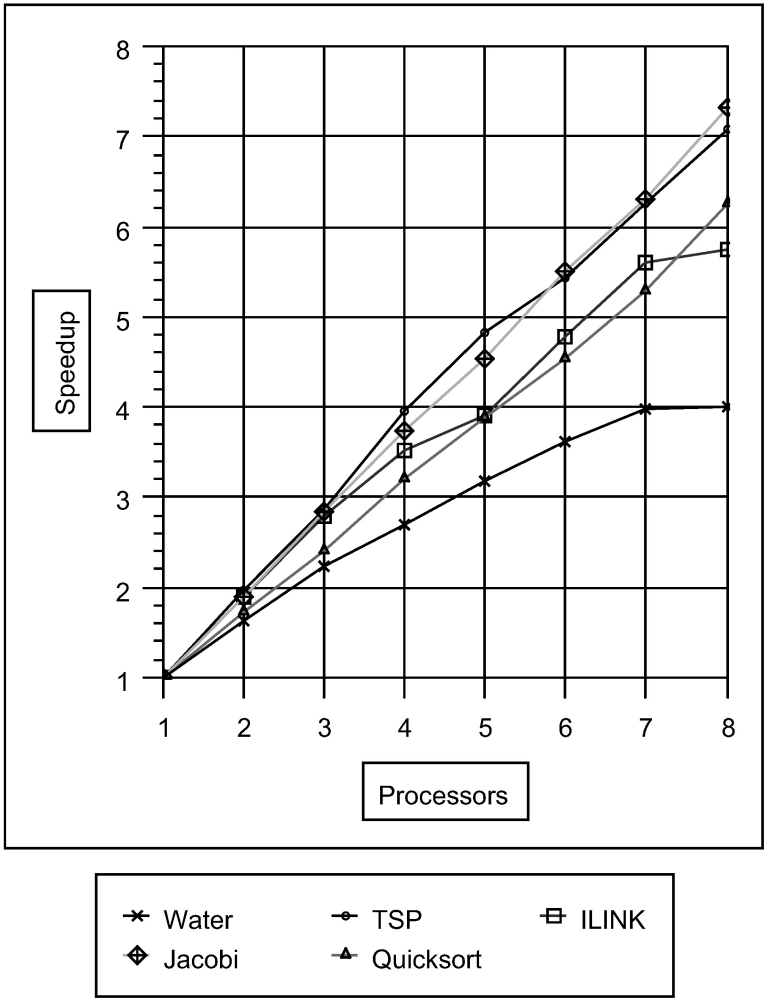
- Applications
  - Water – molecular dynamics simulation
  - Jacobi – Successive Over-Relaxation
  - TSP – branch & bound algorithm to solve the traveling salesman problem
  - Quicksort – using bubblesort to sort subarray of less t han 1K element
  - ILINK – genetic linkage analysis

Evaluation

Execution statistics

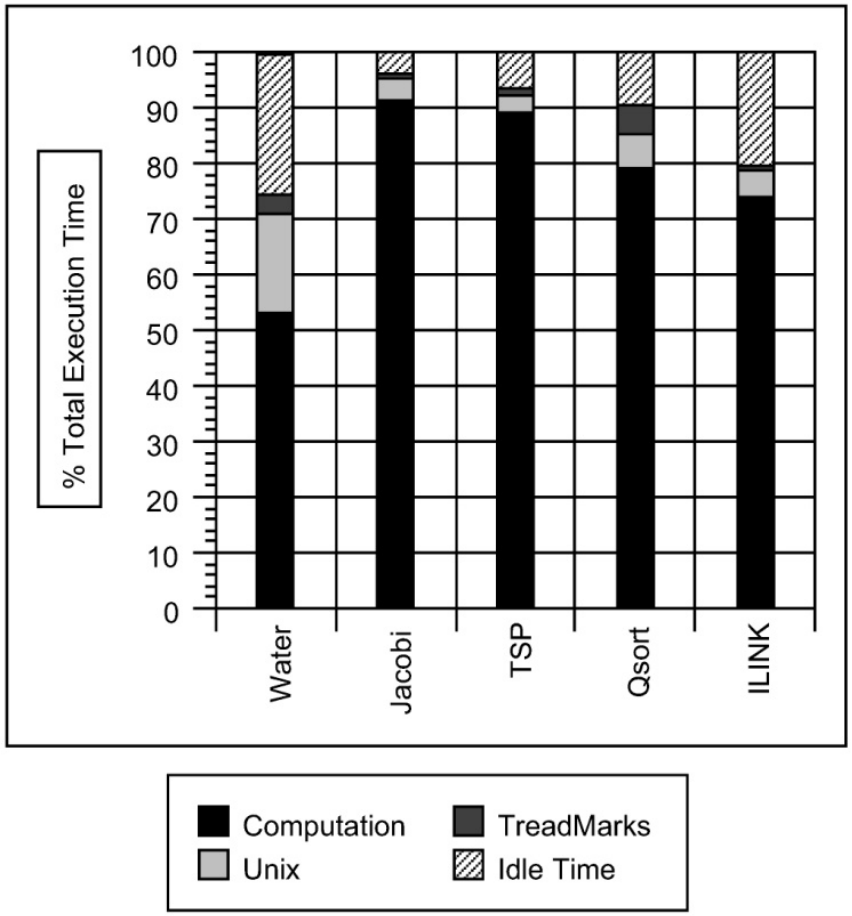
	Water	Jacobi	TSP	Quicksort	ILINK
Input	343 mols 5 steps	2000x1000 floats	19-city tour	256000 integers	CLP
Time (secs)	15.0	32.0	43.8	13.1	1113
Barriers/sec	2.5	6.3	0	0.4	0.4
Locks/sec	582.4	0	16.1	53.9	0
Msgs/sec	2238	334	404	703	456
Kbytes/sec	798	415	121	788	164

Speedup



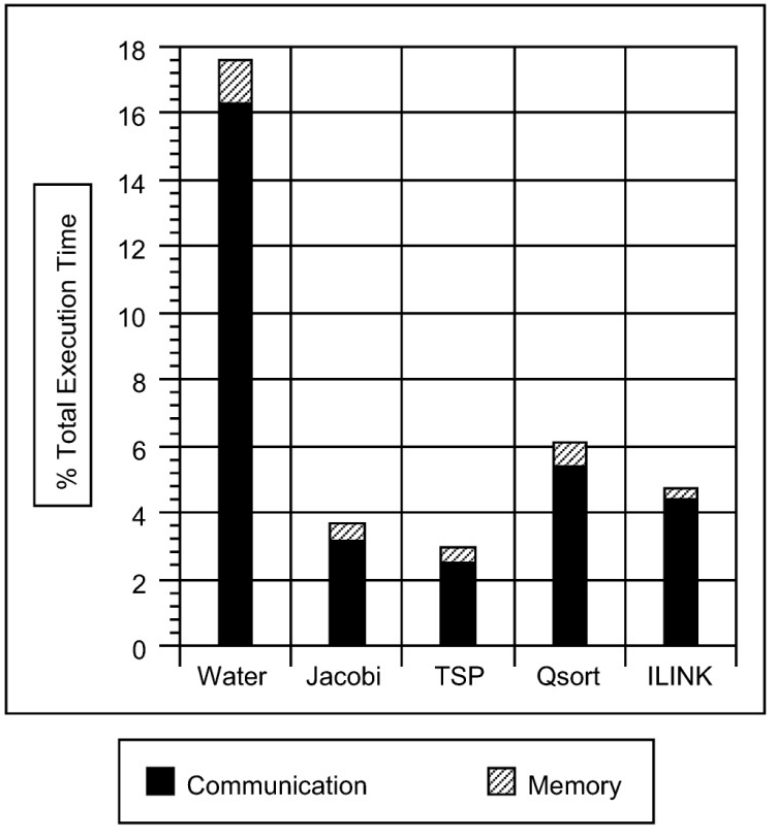
Evaluation

Execution time breakdown

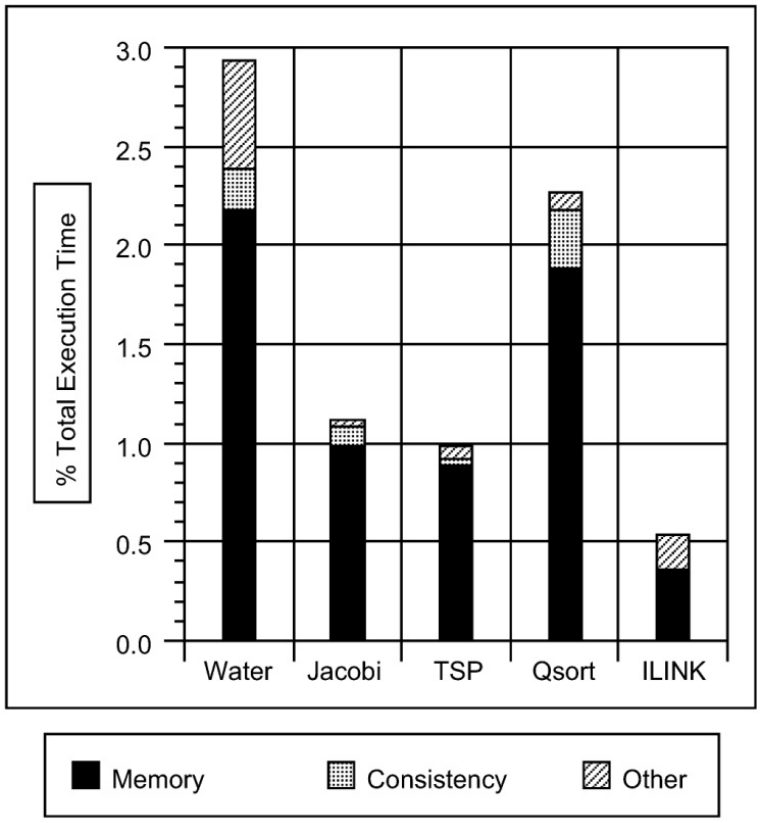


Evaluation

Unix overhead breakdown

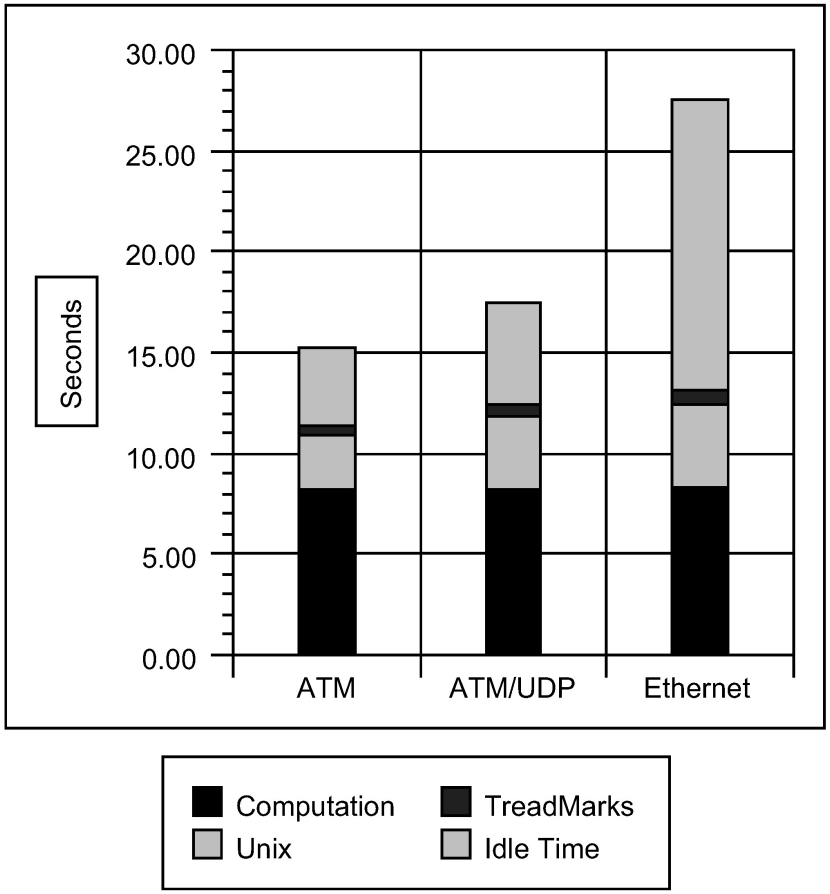


TreadMarks overhead breakdown



Evaluation

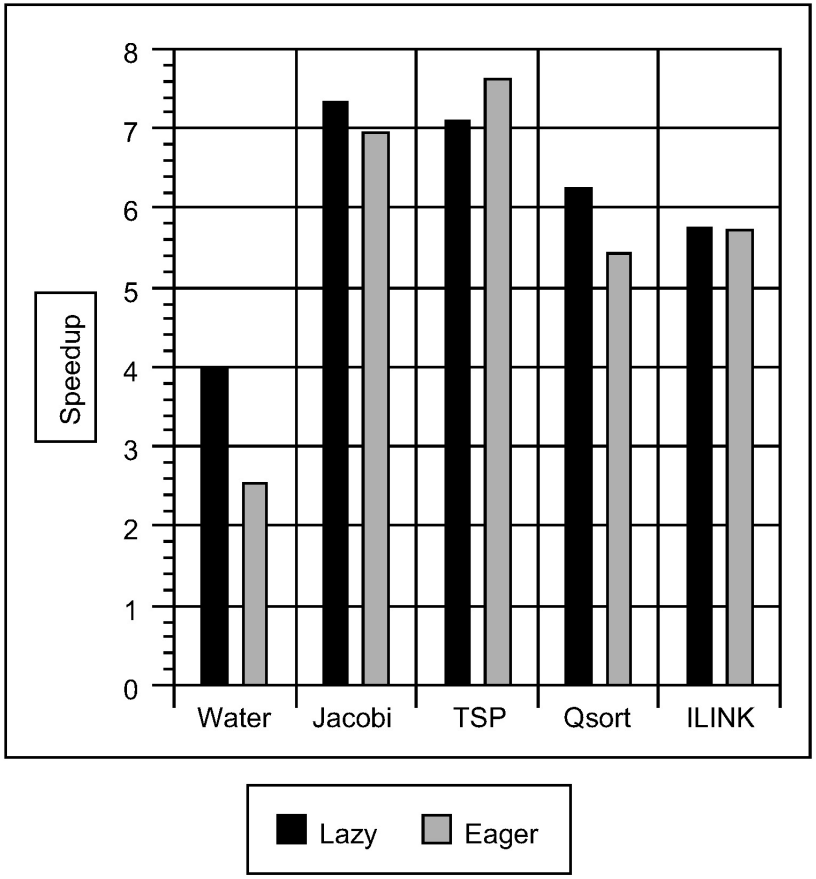
Execution time breakdown for Water



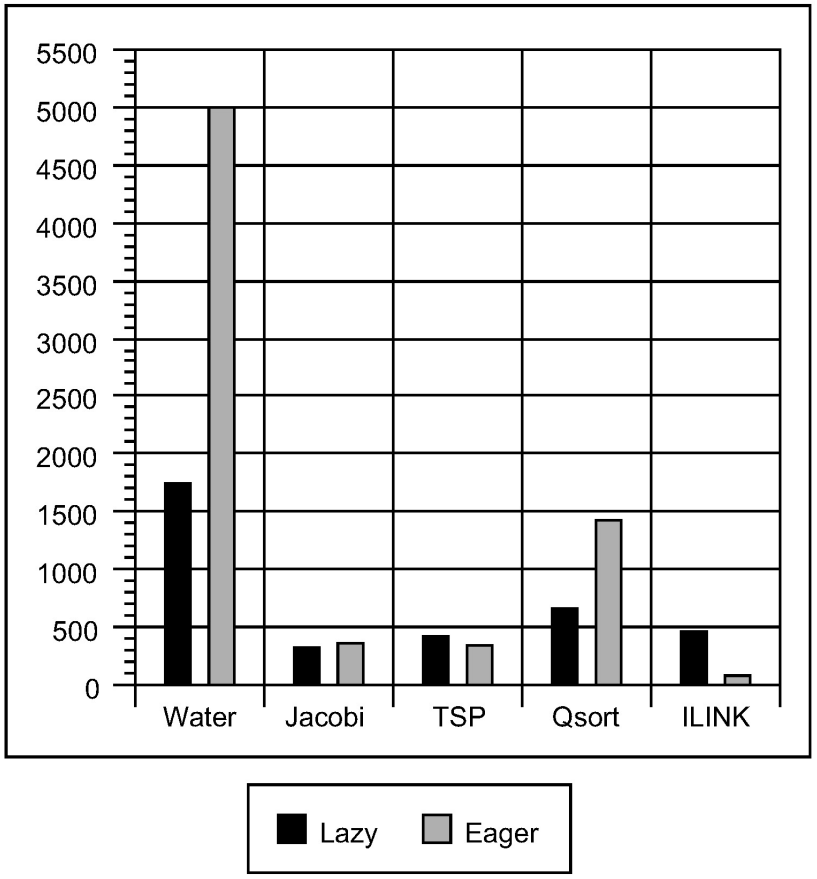
Evaluation

ERC vs. LRC

Speedup



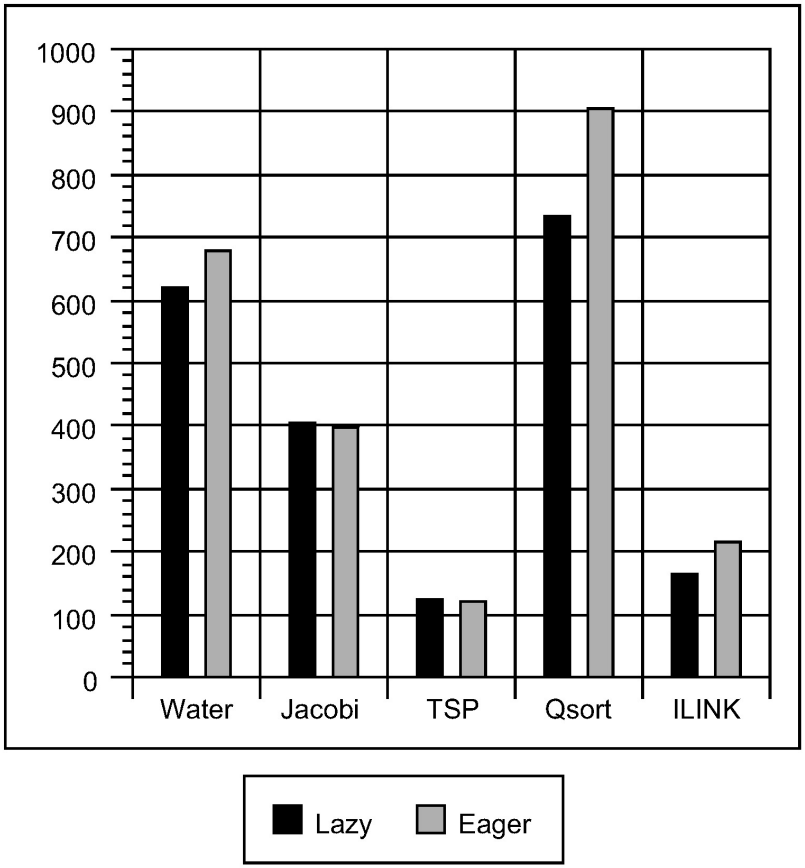
Message rate



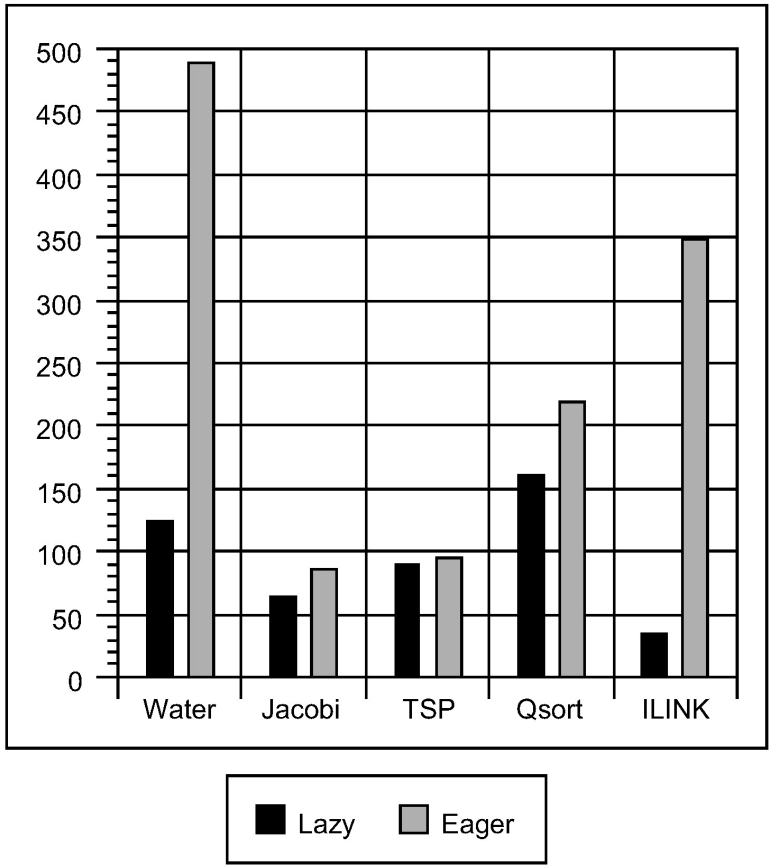
Evaluation

ERC vs. LRC

Data rate



Diff creation rate



Implementation

P0 side

P1 side

pages

. . .	
	P
. . .	
. . .	

Acq(L)

pages

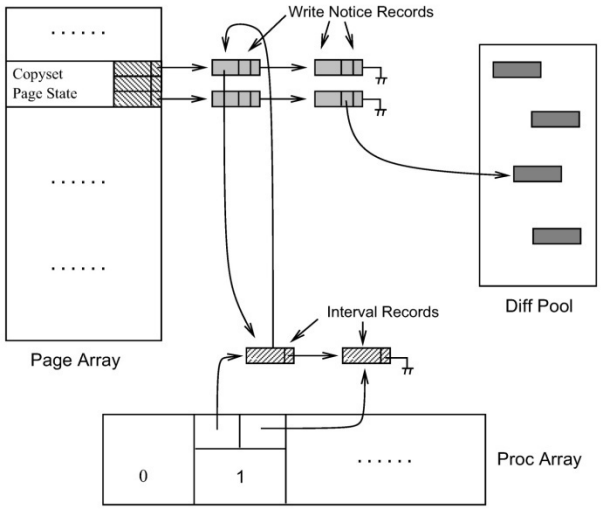
. . .	
	P
. . .	
. . .	

time stamp

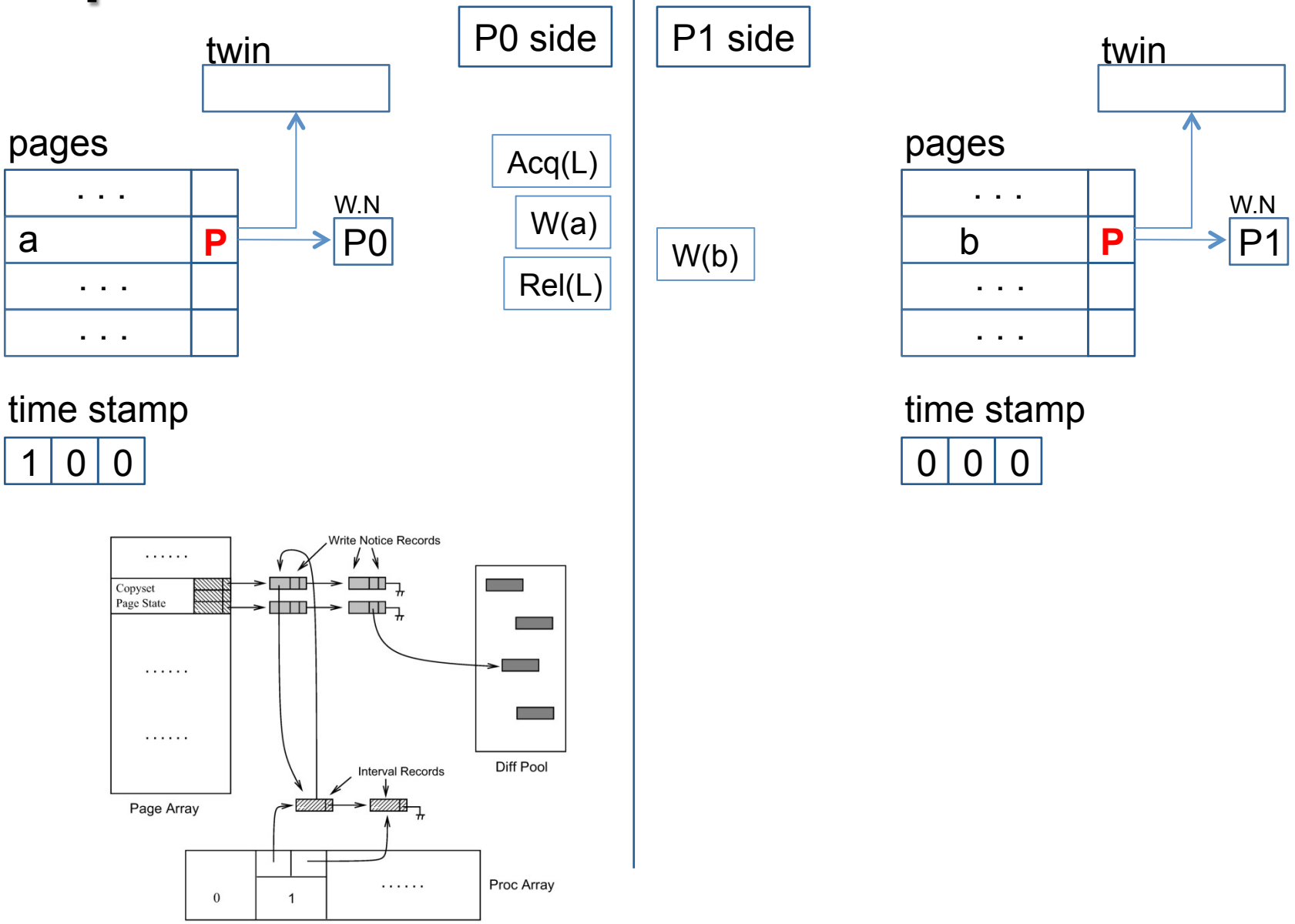
0	0	0

time stamp

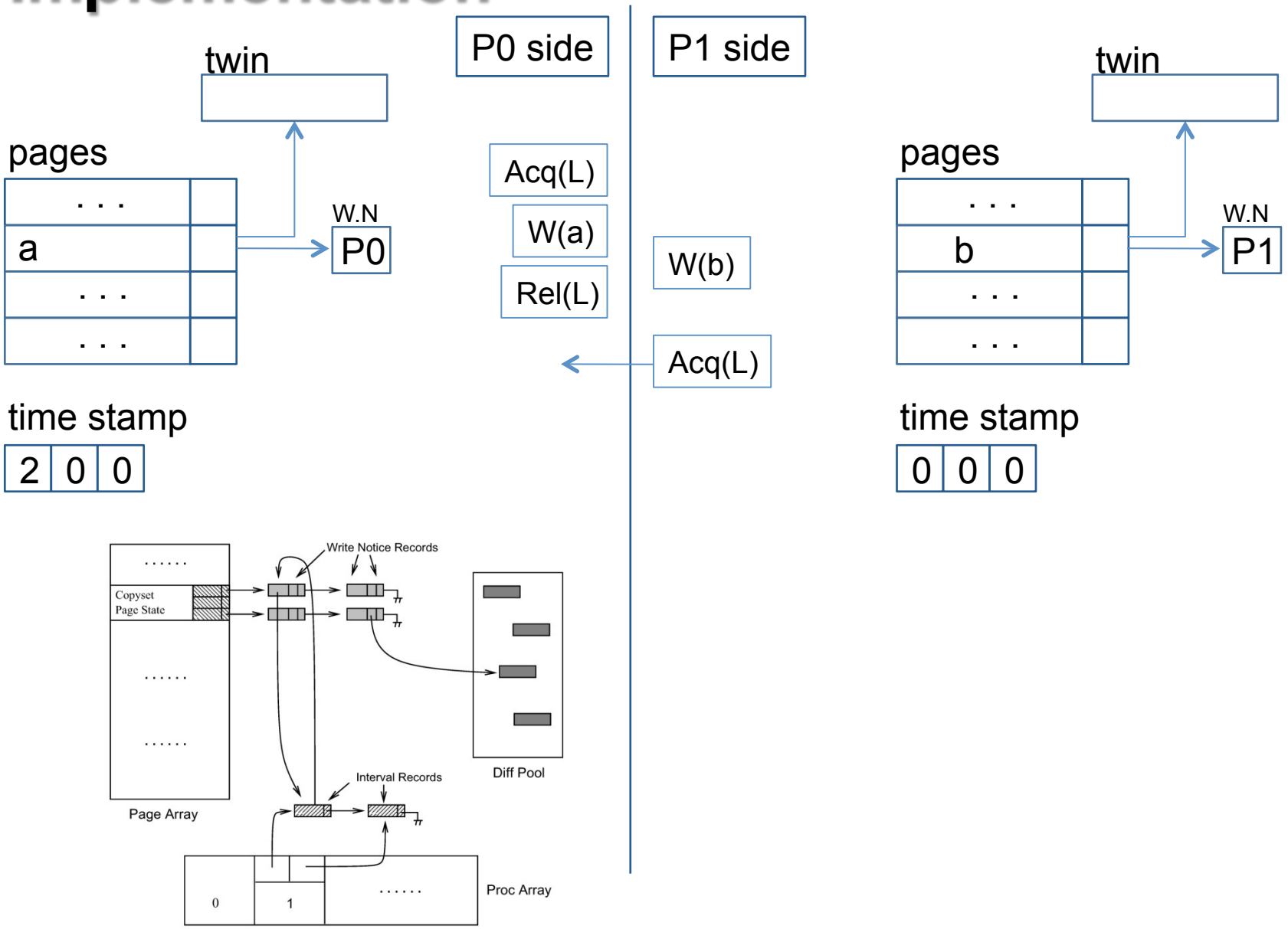
0	0	0



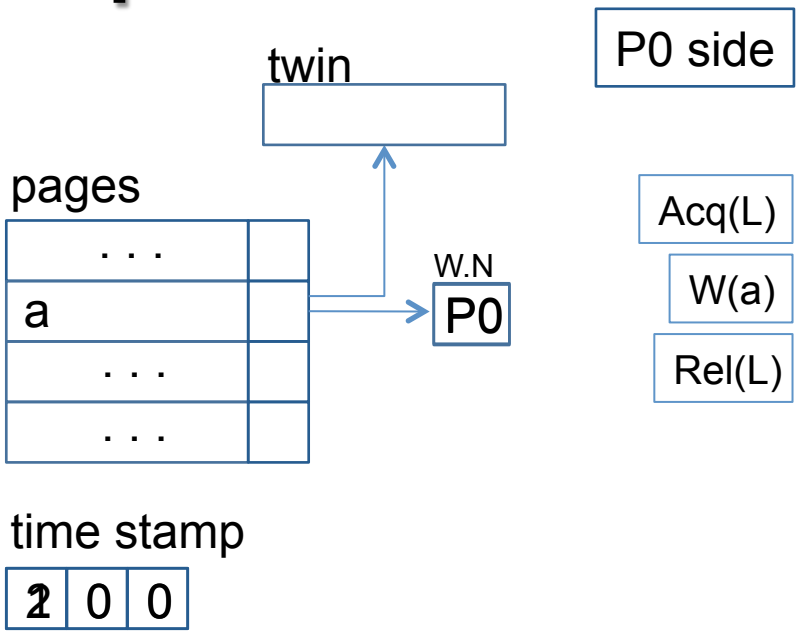
Implementation

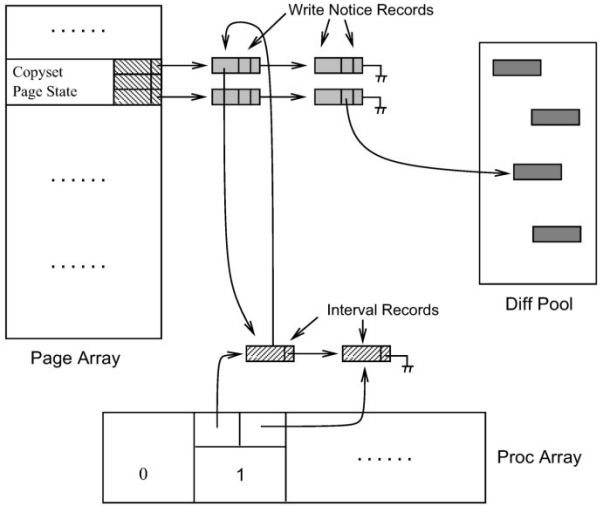


Implementation

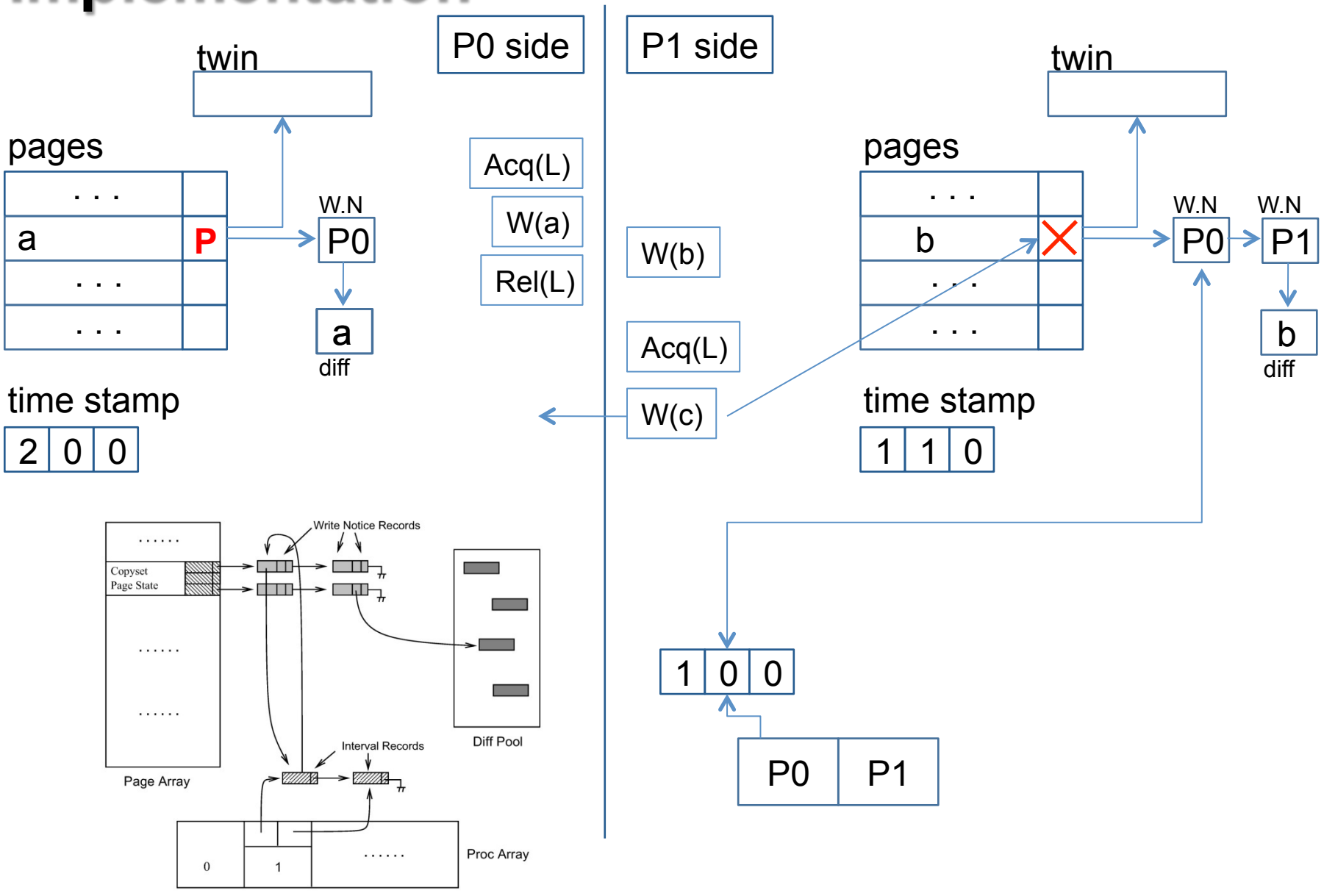


Implementation





Implementation

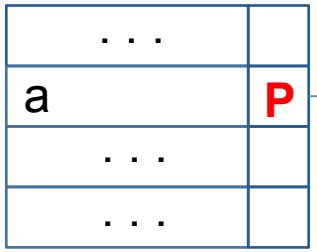


Implementation

P0 side

P1 side

pages



W.N

P0

a

diff

Acq(L)

W(a)

Rel(L)

W(b)

Acq(L)

W(c)

Rel(L)

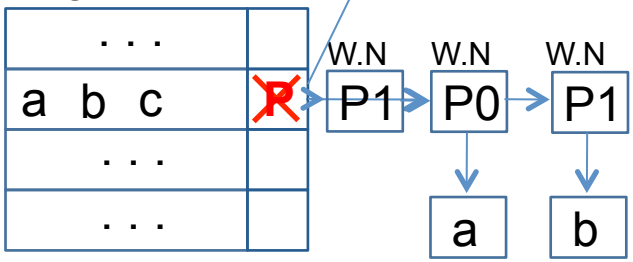
time stamp

2	0	0

twin

a b

pages

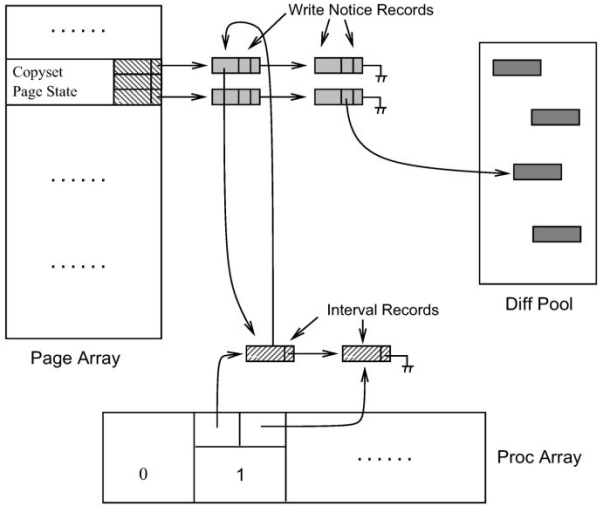


W.N W.N W.N

diff diff

time stamp

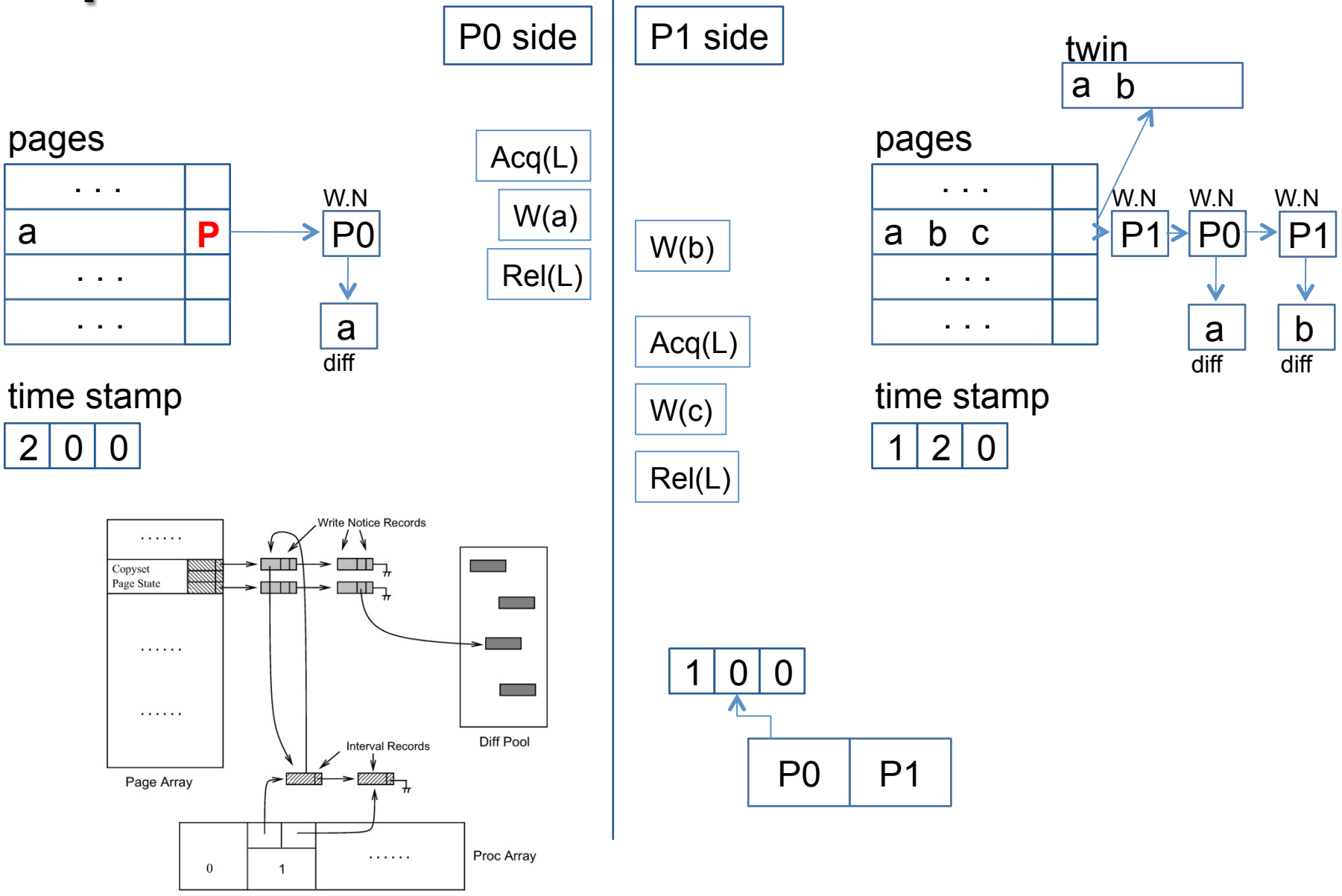
1	1	0



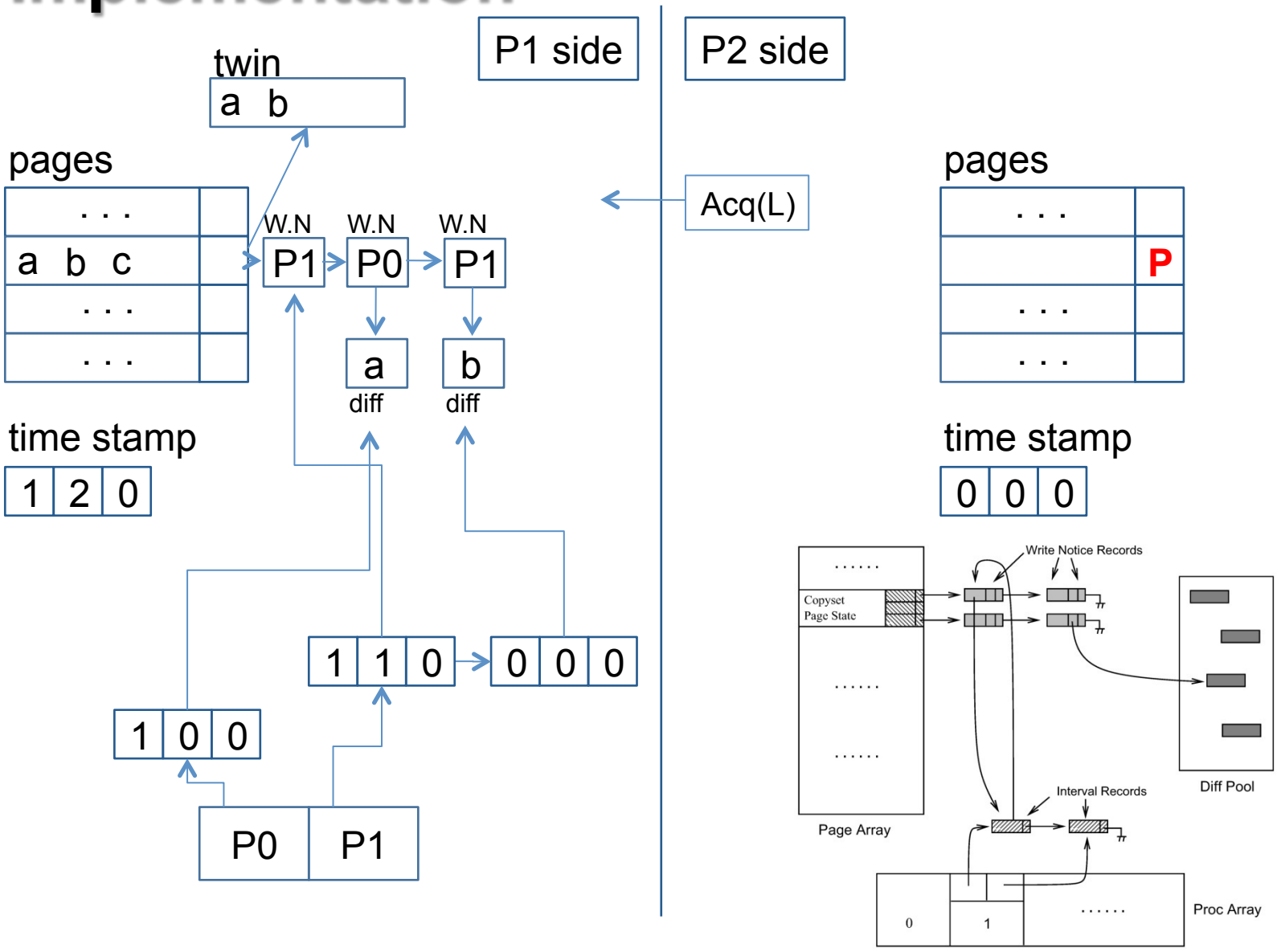
1	0	0

P0	P1

Implementation



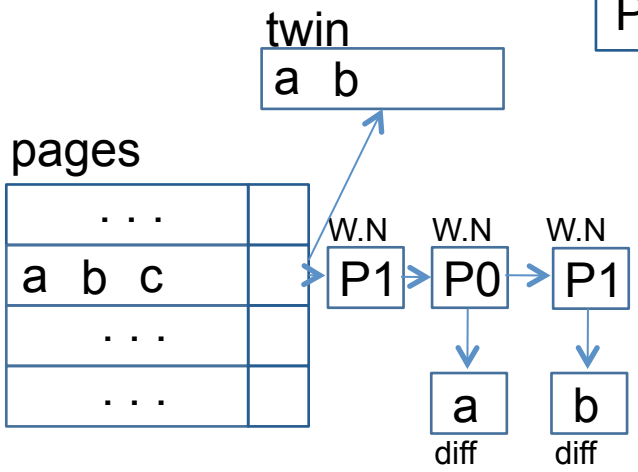
Implementation



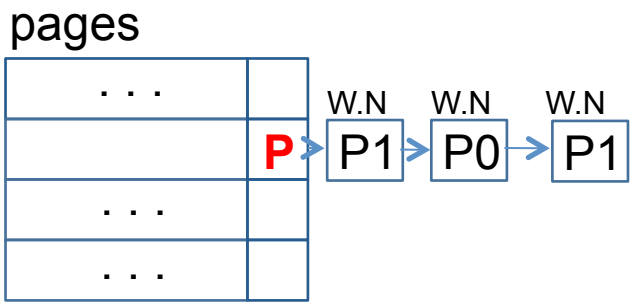
Implementation

P1 side

P2 side



Acq(L)



time stamp

1	2	0

time stamp

1	1	1