

Deadlocks

{
read x
write y

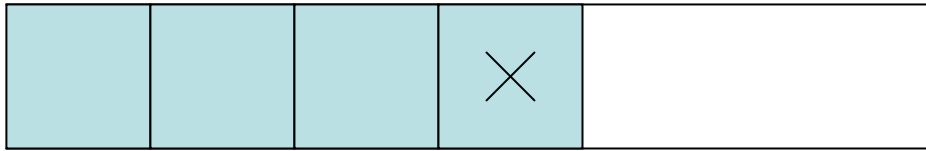
comput'n
COMMIT()

{

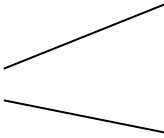
acq l _x	acq l _y
read x	read y
→ acq l _y	→ acq l _x
W y	W x

- 1) Timers
- 2) Waits for graph

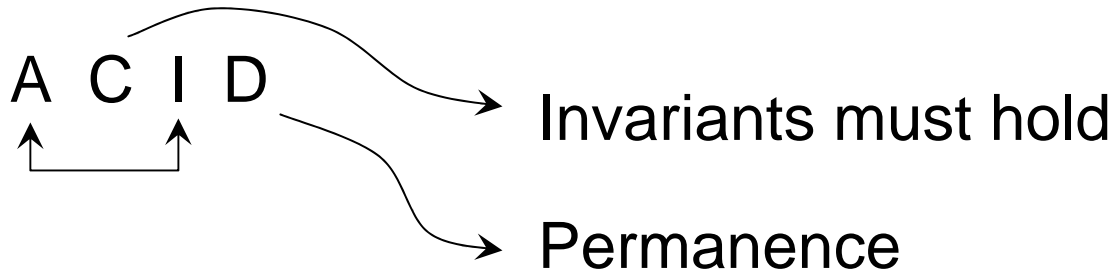
Logs & Locks



Applications

- 1) Transactions 
 - Consistency
 - Durability
- 2) Multi-site atomicity

Transactions



Centralized

Integrity rules

SID	Name	DeptID
35	----	43

Dept ID	Dept name
43	
25	

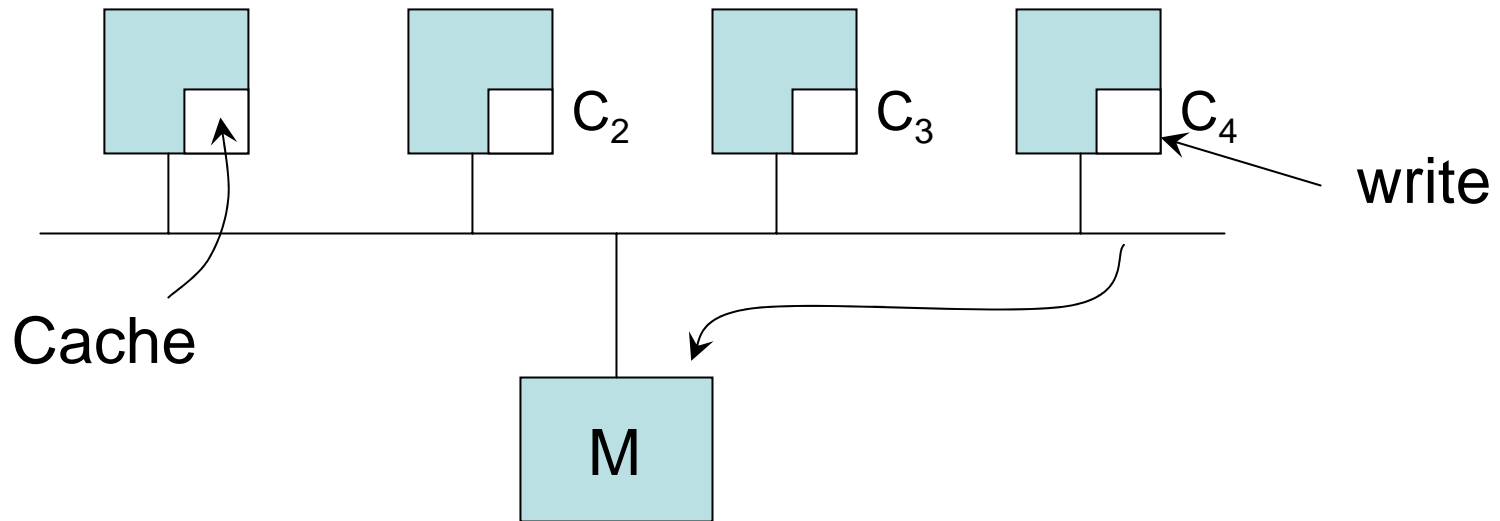
Distributed Data

- DNS → Expiration time
- Web caches
 - ↳ “if-modified-since”

Strong cons:

Read returns result
of last write

Eventual consistency



- 1) Write-thru cache
- 2) Snoopy cache