

MIT OpenCourseWare
<http://ocw.mit.edu>

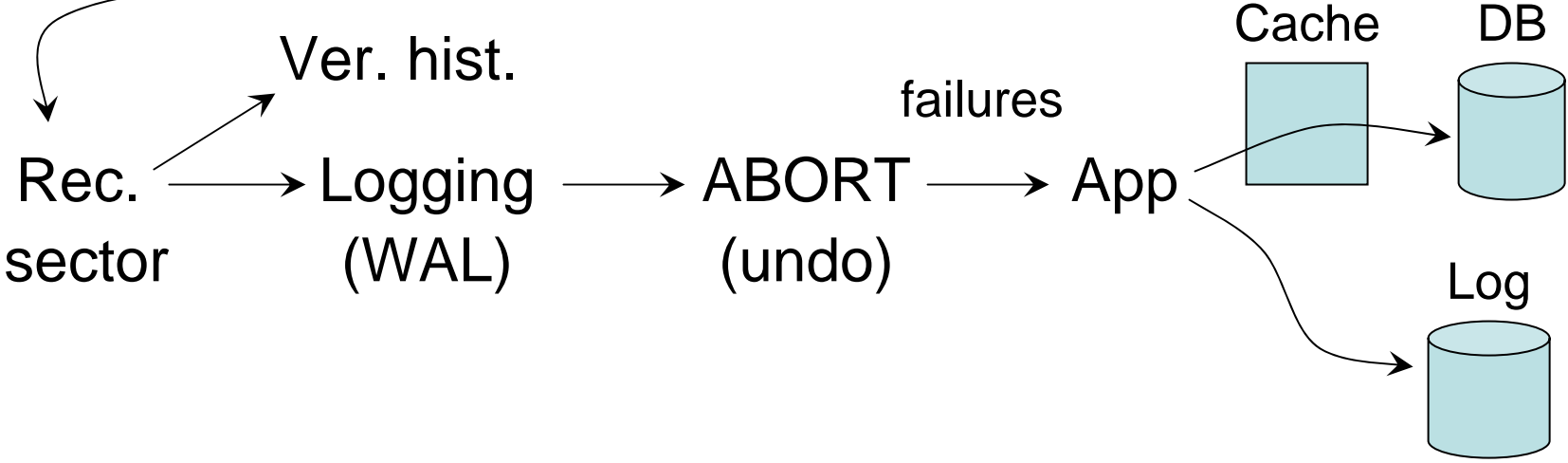
6.033 Computer System Engineering
Spring 2009

For information about citing these materials or our Terms of Use, visit: <http://ocw.mit.edu/terms>.

Atomicity

Isolation

Recoverability



Recovery

- 1) Undo
- 2) Redo

- 1) Scan log backwards
- 2) Winners = C + A
- 3) Losers:
- 4) Redo C winners + undo losers.

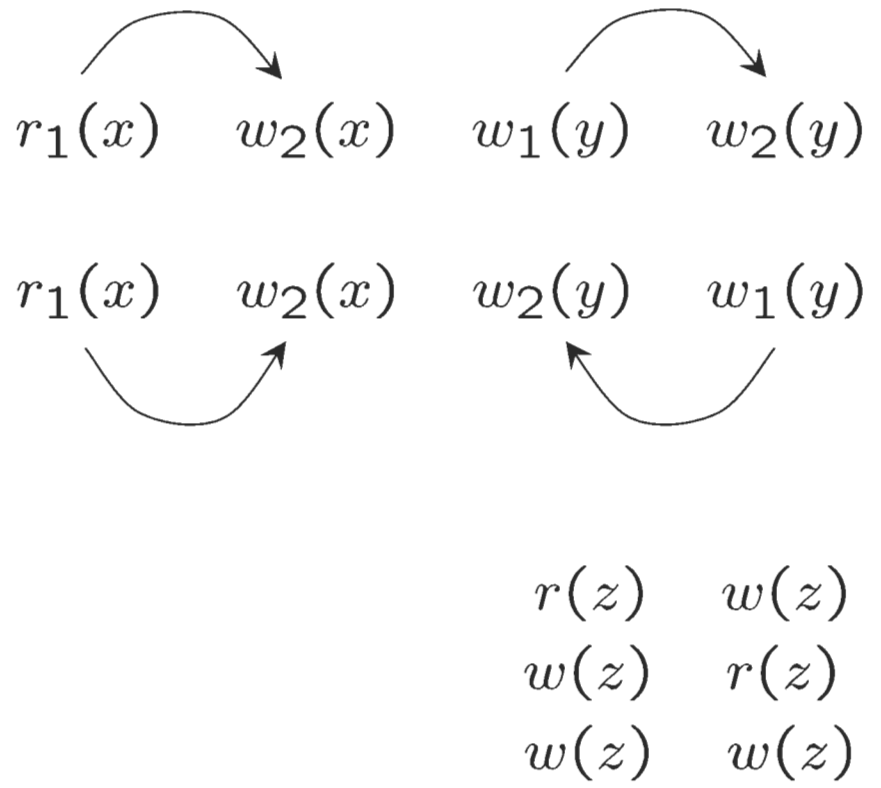
Isolation

T₁

- 1) read x
- 3) write y

T₂

- 2) write x
- 4) write y



Serializability:

Trace's conflict arrow in same order as some serial order of actions

Action Graph

T₁

T₂

T₃

T₄

1) $r_1 x$

2) $w_2 x$

3) $r_3 y$

4) $r_4 x$

5) $w_1 y$

6) $w_2 y$

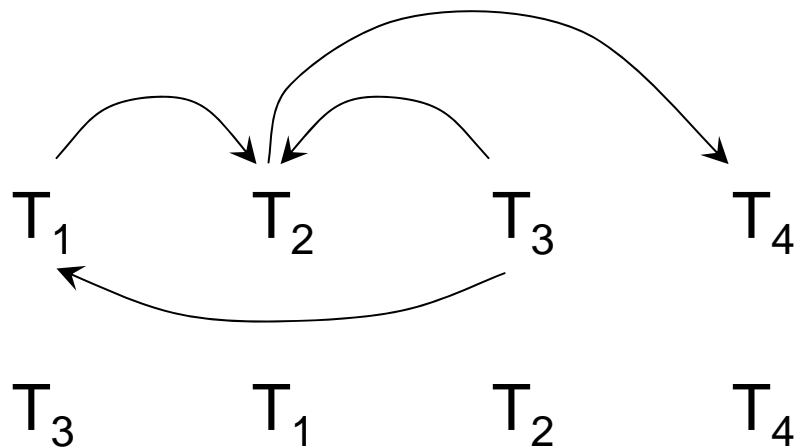
7) $w_3 z$

T_1 T_2

T_1 T_3

T_2 T_3

T_2 T_4



If Action Graph is acyclic
 \Leftrightarrow trace is serializable

\Rightarrow : Topo. sort

Locks:

acq (lock of x)
rel (")

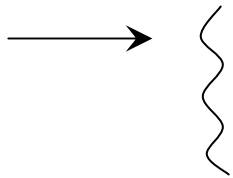
1 {
acq l_x
r₁(x)
rel l_x
3 {
acq l_y
w₁(y)
rel l_y

2 {
acq l_x
w₂(x)
acq l_y
w₂(y)
4 {
rel l_x
rel l_y

Isolation:

acq l_x
acq l_y

Simple
locking

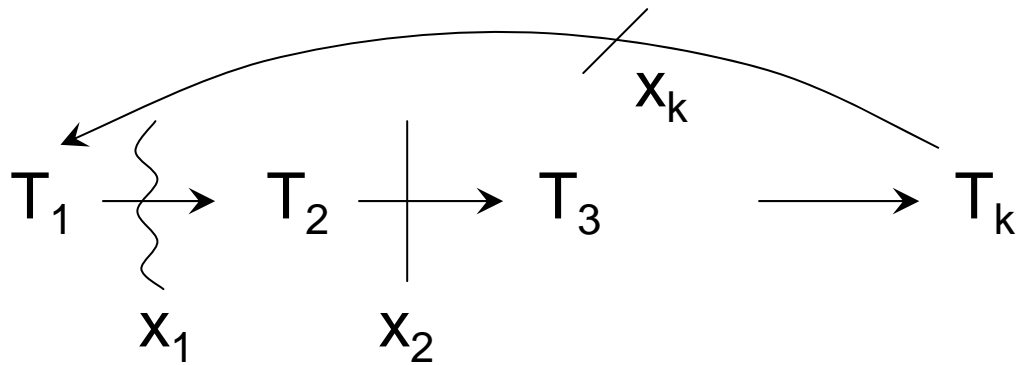


acq l_x
r x
acq l_y
r y

Two-phase locking (2 PL)

No release before ALL acquires

Correct



rel l_1

acq l_1

rel l_2

acq l_2

acq l_{k-1}

rel l_k

acq l_k