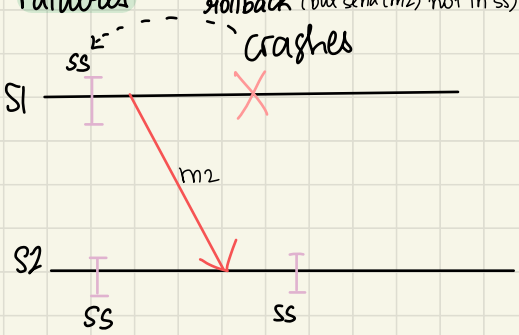
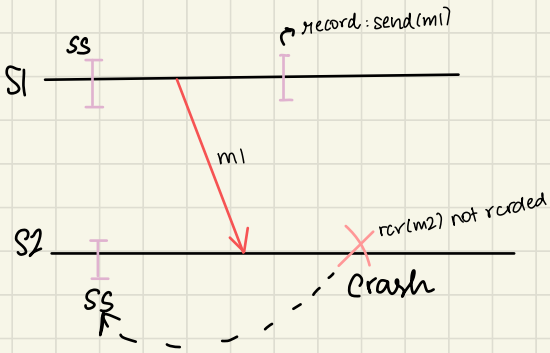
CAT-3

Portions

have to get
Byzantine consensus?

Failures



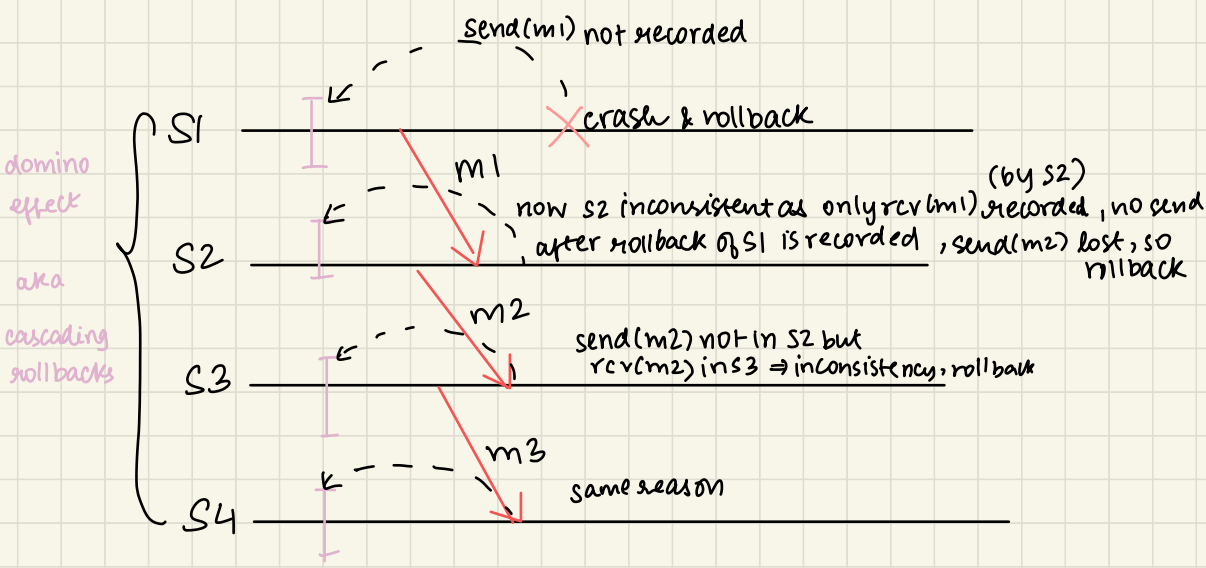


System is consistent

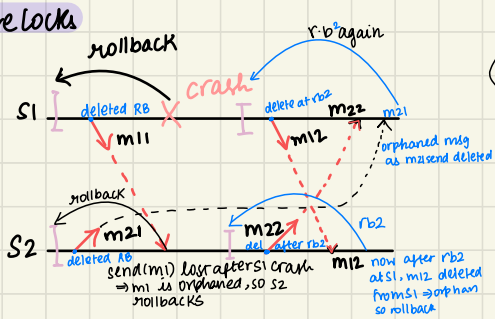
when rollback of S2 happens, it will expect rcv(m2), but it never happens. *

Doesn't affect consistency but affects performance

* ie, S2 assumes m1 in transit still



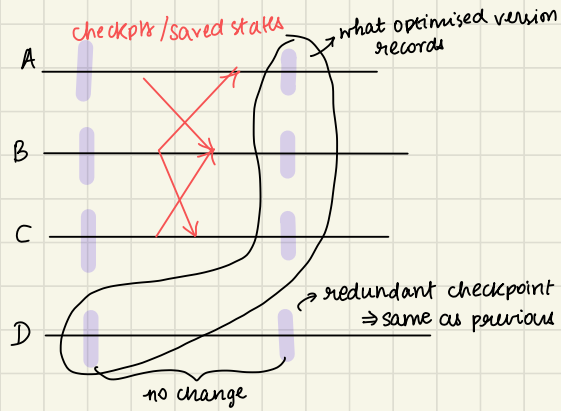
Live locks



rollbacks keep reoccurring.

no progress (1 physical crash leads to many logical crashes)

Sync Checkpointing → only records useful cps (unlike CSRA)



CSRA → maintains consistency

but records unwanted cp-s

only in CSRA

Sync Checkpointing - Algorithm

Initiator doesn't BC marker, only sends through cohorts

LR$_X^Y$: 2 (label of last msg Y rcvd from X)

LR$_Y^X$: 1

FS$_X^Y$: 1 (label of first msg X sent to Y)

FS$_Y^X$: 1

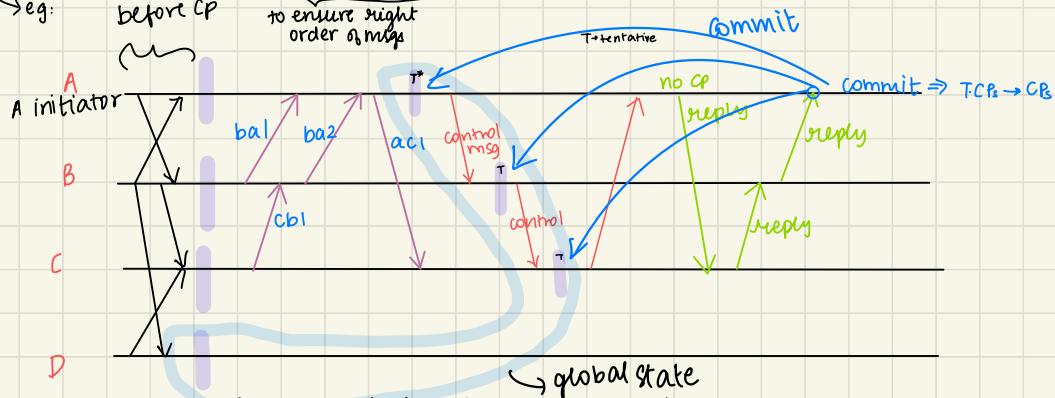
CASE-1

$Cohort_X : \{ Y \mid LR \downarrow_X Y \}$

Y is a cohort of X if X has rcvd a msg from Y

Y will only cp if $\underbrace{LR \downarrow_X^Y \geq FS \nearrow_Y^X}_{\text{to ensure right order of msgs}}$ AND $FS \nearrow_Y^X > 0$

eg:



after A cps, A sends M to its cohorts

here $cohort_A = \{b\}$

$\boxed{\text{So, M only sent to B from A}}$

after B rcvs control msg,

check if condition holds,

$LR \downarrow_A^B \geq FS \nearrow_B^A$ AND $FS \nearrow_B^A > 0$

2 ≥ 1 True 1 > 0 True

since true, B cp.s & sends M to cohorts

$cohort_B = \{C\}$

after C rcvs control msg, check condition

$LR \downarrow_B^C \geq FS \nearrow_C^B$ AND $FS \nearrow_C^B > 0$

1 ≥ 1 true 1 > 0 true

⇒ C cps & send M to cohorts

$cohort_C = \{A\}$

after A rcvs cp, check if

$LR \downarrow_C^A \geq FS \nearrow_A^C$ AND $FS \nearrow_A^C > 0$

here, A already took cp, so no msg sent after that CP

⇒ $FS \downarrow_C^A = 0$ after T. CP

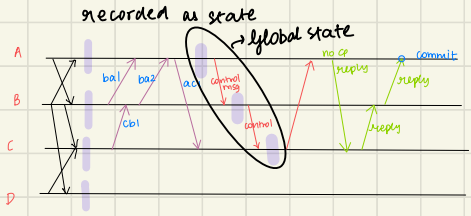
so 1 >= 0 true but 0 > 0 false

A now doesnt take CP but replies to C to ack.

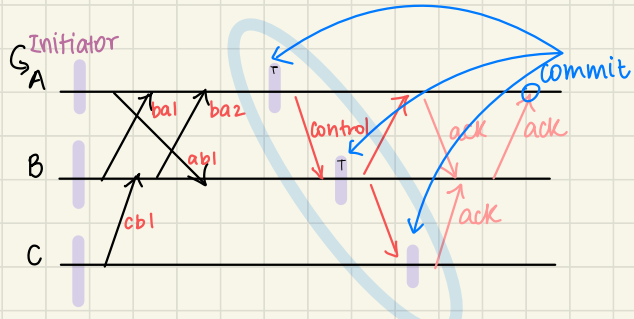
then C replies to B & B to A

after Initiator rcvs a reply, it commits

After commit, tentative cp becomes recorded as state



CASE 2



at A → initiator

$Coh_A = \{B\}$

$LR_{A}^{\swarrow B} = ba2 \to 2$ (msg B to inform)

at B

rcv msg from A

$LR_{A}^{\swarrow B} \geq FS_{B}^{\nearrow A}$ && $FS_{B}^{\nearrow A} > 0$

$2 >= 1$ && $1 > 0$

checkpoint

$Coh_B = \{C, A\}$

A already cped, but C haunt

$LR_{B}^{\swarrow A} = 1$ (send A)

$LR_{B}^{\swarrow C} = 1$ (send C)

concurrently

at C

check

$LR_{B}^{\swarrow C} \geq FS_{C}^{\nearrow B}$ &&

$FS_{C}^{\nearrow B} > 0$

$1 \geq 1$ ✓ $1 > 0$ ✓

checkpoint

$coh_C = \{\Phi\}$

terminate & ack B

at A

check

$LR_{B}^{\swarrow A} \geq FS_{A}^{\nearrow B}$ & $FS_{A}^{\nearrow B} > 0$

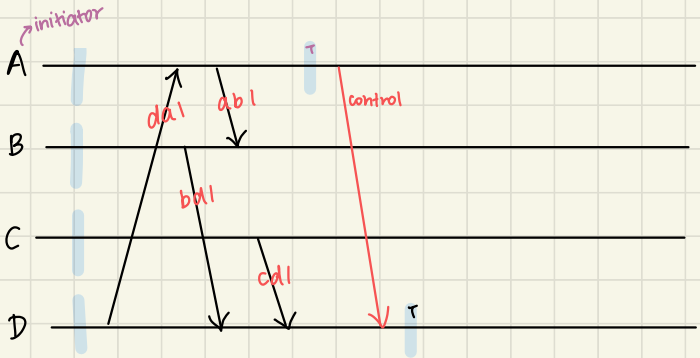
only consider after tentative

cp of A ⇒ $FS_{A}^{\nearrow B}$ after cp = 0

$1 \geq 0$ X no cp.

ack B & terminate

example



at A

$con_A = \{D\}$

$LR_A^{\swarrow D} = 1$ (send to D)

at D

$LR_A^{\swarrow D} \geq FS_D^{\nearrow A}$ && $FS_D^{\nearrow A} > 0$

$1 \geq 1$ ✓ $1 > 0$ ✓

checkpoint

$con_D = \{C, B\}$

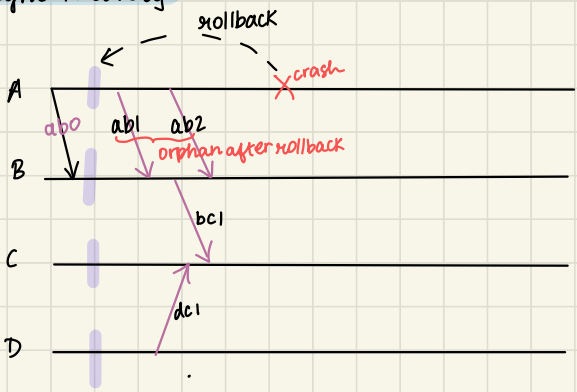
$LR_D^{\swarrow C} = 1$ $LR_D^{\swarrow B} = 1$

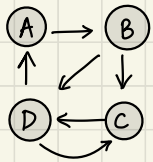
↳ send to C ↳ send to B

Concurrently

at C at B

Sync Recovery





NOTE: alerts sent through all outgoing channels

$LS_A^{\nearrow B} \rightarrow ab1 = 1$

$LR_B^{\swarrow A} \rightarrow ab1 = 1$

After A's crash, B has to become aware of it. How?

↳ A can inform B after it recovers, (if B is A's cohort) → Case 1

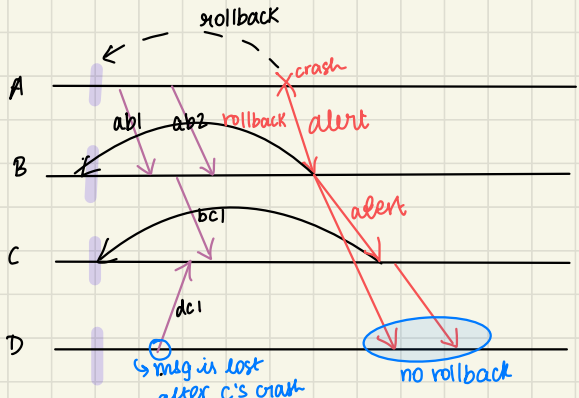
↳ If A can't recover, (2nd case)

For B's recovery

① $LR_B^{\swarrow A} > LS_A^{\nearrow B} \Rightarrow$ B has to recover

(last sent A to B after A's crash)

$1 > 0 \Rightarrow$ B has to recover, ie, rollback to cp. (send alert to C & D)



but this doesn't affect consistency, only performance

② a

at C, (from B)

$LR_C^{\swarrow B} > LS_B^{\nearrow C}$?

$1 > 0 \rightarrow$ true

⇒ C has to recover

x rollback. (alert D)

② b

concurrently rcv alert from B

at D (from B)

$LR_D^{\swarrow B} > LS_B^{\nearrow D}$?

$0 > 0 \rightarrow$ false

no rollback

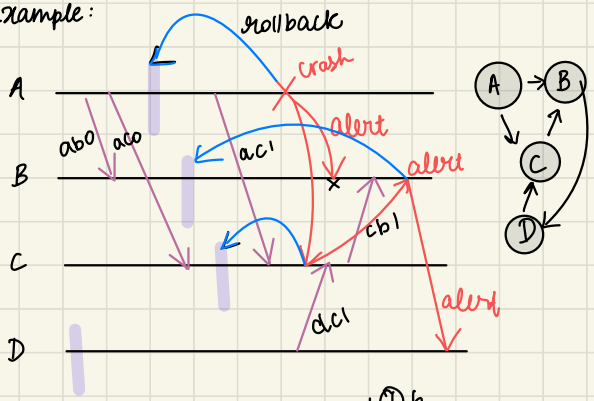
③ at D (from C)

$LR_D^{\swarrow C} > LS_C^{\nearrow D}$?

$0 > 0 \rightarrow$ false

no rollback

example:



① a at B (alert from A)

$LR_B^{\swarrow A} > LS_A^{\nearrow B}$?

0 > 0 → false

no rollback

① b at C (alert from A)

$LR_C^{\swarrow A} > LS_A^{\nearrow C}$?

1 > 0 (after crash) → true

rollback & alert B.

② at B (alert from C)

$LR_B^{\swarrow C} > LS_C^{\nearrow B}$?

1 > 0 → true

rollback & alert D

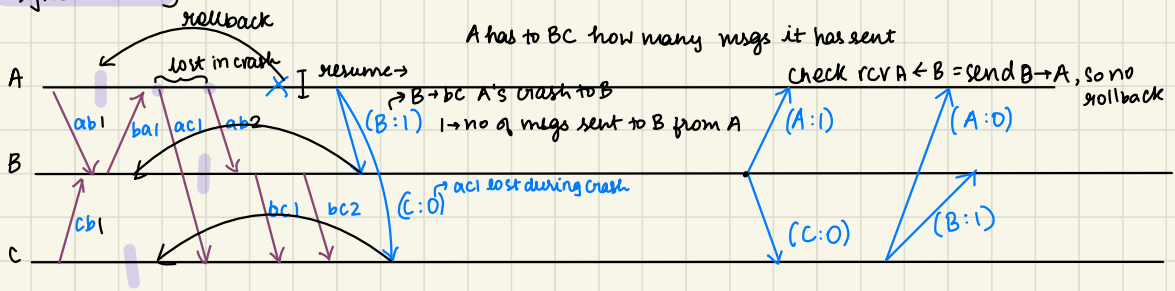
③ at D (alert from B)

$LR_D^{\swarrow B} > LS_B^{\nearrow D}$?

0 > 0 → false

no rollback

dsync Recovery



at B,

send A→B = 1

recv B←A = 2

send < recv ⇒ rollback (but to where?)

↳ not to its prev cp as there is no coordination b/w checkpts

till any pt before rcv of msg ab2

at C

send A→C = 0

rcv C←A = 1

rcv ? send ⇒ rollback