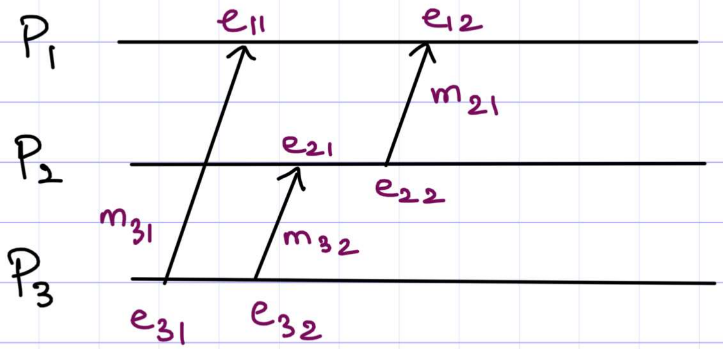
Causal Ordering of Messages - Non Broadcast



send (m_{31}) → send (m_{32})

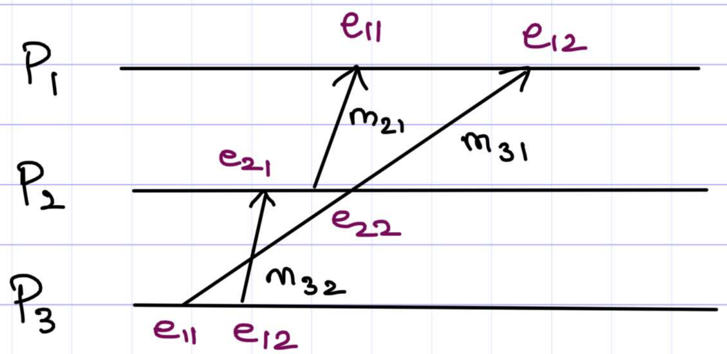
send (m_{32}) → rec (m_{32})

rec (m_{32}) → send (m_{21})

send (m_{21}) → rec (m_{21})

Conclusion } ⇒ rec(m_{31}) → rec(m_{21})

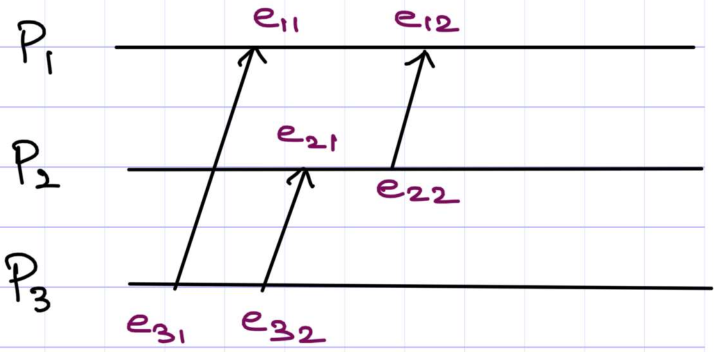
Delivery is causal.



Non-causal delivery.

m_{21} should be delivered after m_{31}.

Schipper - Eggli - Sandoz Protocol



$\underline{e_{31}}$

$C(0,0,1)$

$V_3(x, x, x)$

Send t, V_3.

Update V_3.

$V_3((0,0,1), x, x)$

t - timestamp

V_m - message vector

$\underline{e_{32}}$

$C(0,0,2)$ $V_3((0,0,1), x, x)$

Send t, V_3.

Update V_3.

$V_3((0,0,1), (0,0,2), x)$

$\underline{e_{11}}$

$C(0,0,0)$ $V_1(x, x, x)$

$t(0,0,1)$ $V_m(x, x, x)$ ← received.

→ Both null.

⇒ nothing has happened yet.

Accept the message.

$V_1(x, x, (0,0,1))$

copied here because message received from P_3.

Merge the vectors & copy the timestamp to the vector.

$C(0,0,1)$

e_{21}

$C(0,0,0) \quad V_2(x, \boxed{x}, x)$ no conflict

$t(0,0,2) \quad V_m((0,0,1), \boxed{x}, x)$

$V_2((0,0,1), x, (0,0,2)) \quad C(0,0,2)$

e_{22}

$C(0,1,2) \quad V_2((0,0,1), x, (0,0,2))$

Send C, V_2.

Update V_2.

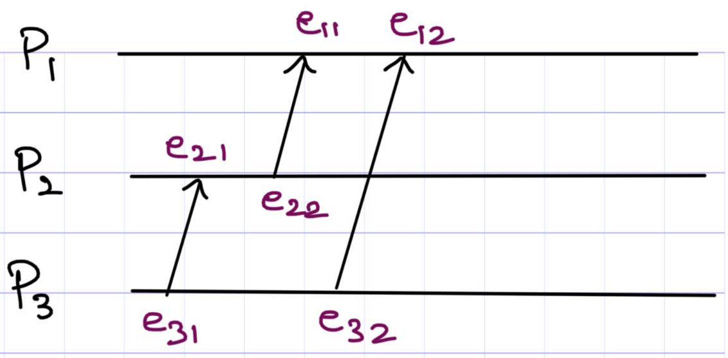
$V_2((0,1,2), x, (0,0,2))$

e_{12}

$C(0,0,1) \quad V_1(x, x, (0,0,1))$

$t(0,1,2) \quad V_m((0,0,1), x, (0,0,2))$

$\underline{e_{31}}$



Send event
$C_3\,(0, 0, 1)$
$V_3\,(x, x, x)$

Send C_3, V_3.
Update V_3.
$V_3\,(x,\ (0, 0, 1),\ x)$.

$\underline{e_{32}}$

Send event
$C_3\,(0, 0, 2)$
$V_3\,(x,\ (0, 0, 1),\ x)$

Send C_3, V_3.
Update V_3.
$V_3\,((0, 0, 2),\ (0, 0, 1),\ x)$.

$\underline{e_{21}}$

$C_2\,(0, 0, 0)$ $V_2\,(0, 0, 0)$

Receive
$C_m(0, 0, 1)$ & $V_m(x, x, x)$. ⇒ First msg. from sender.

Accept the message.

Update V_2.
$V_2\,(x,\ x,\ (0, 0, 1))$

Merge Clock Values.
$C_2\,(0, 0, 1)$

$\underline{e_{22}}$

Send event

$C_2\,(0, 1, 1)$

$V_2\,(x, x, (0, 0, 1))$

Send C_2 & V_2.

Update V_2.

$V_2\,((0, 1, 1), x, (0, 0, 1))$

$\underline{e_{11}}$

$C_1\,(0, 0, 0)$ $V_1\,(\boxed{x}, x, x)$

no conflict.

Receive

$Cm\,(0, 1, 1)$ $Vm\,(\boxed{x}, x, (0, 0, 1))$

Accept the message.

Update C_1. $C_1\,(0, 1, 1)$

Update V_1. $V_1\,(x, \underbrace{(0, 1, 1)}_{\downarrow}, \underbrace{(0, 0, 1)}_{\downarrow})$

timestamp copied to sender's location.

+ merged V_m & V_1.

now P_1 is aware that P_3 has sent a msg. to P_2.

$\underline{e_{12}}$

$C_1\,(0, 1, 1) \qquad V_1\,(x, (0, 1, 1), (0, 0, 1))$

no conflict

Receive

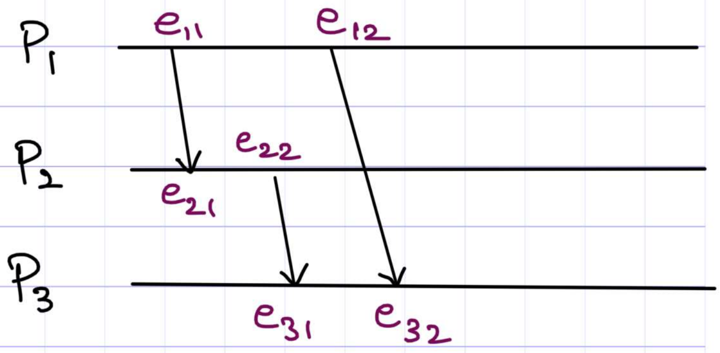
$C_m\,(0, 0, 2) \qquad V_m\,(x, (0, 0, 1), x)$

Accept the message.

Update/merge C_1 & V_1.

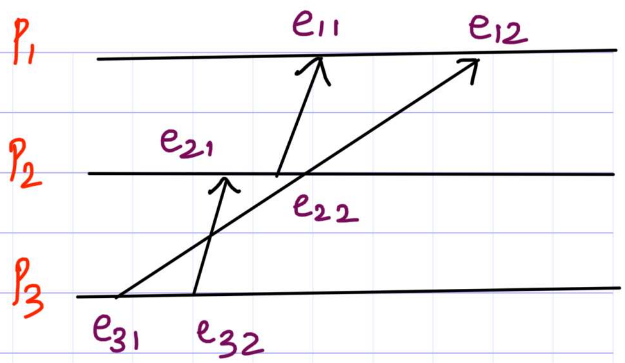
$V_1\,(x, (0, 1, 1), (0, 0, 2))$

$C_1\,(0, 1, 2)$



Try.

Example



e_{31}

C_3 [0 0 0]

V_3 [x x x]

send C_3 [0 0 1] ; V_3 [x x x] to e_{12}.

update C_3 [0 0 1] ; V_3 [(0 0 1) x x]

e_{32}

C_3 [0 0 1]

V_3 [(0 0 1) x x]

send C_3 [0 0 2] ; V_3 [(0 0 1) x x] to e_{21}.

update C_3 [0 0 2] ; V_3 [(0 0 1) (0 0 2) x]

e_{21}

C_2 [0 0 0] V_2 [x x x]

recv [0 0 2] ; V_m [(0 0 1) x x]

update C_2 [0 0 2] ; V_2 [(0 0 1) (0 0 2) x]

e_{22}

C_2 [0 0 2] ; V_2 [(0 0 1) (0 0 2) x]

send C_2 [0 1 2] ; V_2 [(0 0 1) (0 0 2) x]

update C_2 [0 1 2] ; V_2 [(0 1 2) (0 0 2) x]

e_{11}

C_1 [0 0 0] V_1 [x x x]

recv C_1 [0 1 2] V_m [(0 0 1) (0 0 2) x]

send C_1 [0 1 2] V_m [(0 0 1) (0 0 2) x]

to buffer.

$\underline{e_{12}}$

C_1 [0 0 0] V_1 [x x x]

recv. [0 0 1] V_m [x x x]

update C_1 [0 0 1] V_1 [(0 0 1) x x]

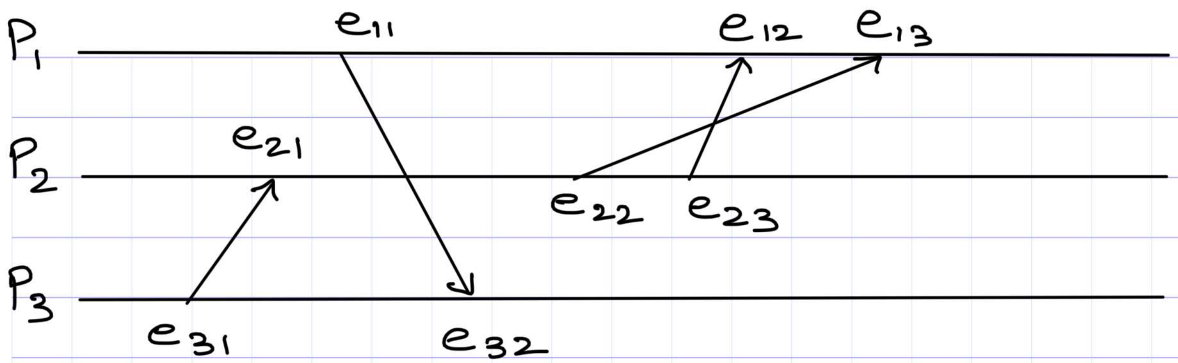
since clock C_1 is updated, check buffer.

$\underline{e_{11}'}$

C_1 [0 0 1] V_1 [(0 0 1) x x]

recv. [0 1 2] V_m [(0 0 1) (0 0 2) x]

update C_1 [0 1 2] V_1 [(0 0 1) (0 0 2) x]



$\underline{e_{31}}$ (send)

$C_3[0, 0, 0]$ $V_3[x, x, x]$

$C_3[0, 0, 1]$ $V_3[x, x, x]$

$\underline{\text{send to } e_{21}}$

$V_3[x, [0, 0, 1], x]$

$\underline{e_{21}}$ (receive)

$C_2[0, 0, 0]$ $V_2[x, \textcircled{x}, x]$

→ $C_m[0, 0, 1]$ $V_m[x, \textcircled{x}, x]$

unset

Accept delivery.

$C_2[0, 0, 1]$ $C_2[x, [0, 0, 1], x]$

$\underline{e_{11}}$ (send)

$C_1[0, 0, 0]$ $V_1[x, x, x]$

$C_1[1, 0, 0]$ $V_1[x, x, x]$

$\underline{\text{send to } e_{32}}$

$V_1[x, x, [1, 0, 0]]$

e_{32} (receive)

$C_3 [0, 0, 1]$ $V_3 [x, [0,0,1], x]$

$C_m [1, 0, 0]$ $V_m [x, x, x]$

unset

Accept delivery

$C_3 [1, 0, 1]$

$V_3 [x, [0,0,1], [1,0,0]]$

e_{22} (send)

$C_2 [0, 0, 1]$ $V_2 [x, [0,0,1], x]$

$C_2 [0, 1, 1]$ $V_2 [x, [0,0,1], x]$

send to e_{13}

$V_2 [[0,1,1], [0,0,1], x]$

e_{23} (send)

$C_2 [0, 2, 1]$ $V_2 [[0,1,1], [0,0,1], x]$

send to e_{12}

$V_2 [[0,2,1], [0,0,1], x]$

e_{12} (receive)

$C_1 [1, 0, 0]$ $V_1 [x, x, [1,0,0]]$

$C_m [0, 2, 1]$ $V_m [[0,1,1], [0,0,1], x]$

buffer it.

e_{13} (receive)

$C_1 [1, 0, 0]$ $V_1 [x, x, [1,0,0]]$

$C_m [0, 1, 1]$ $V_m [x, [0,0,1], x]$

accept delivery

$C_1 [1, 1, 1]$ $V_1 [[0,1,1], x, [1,0,0]]$

$\underline{e_{12}}'$ (redelivery)

$C_1\ [1,1,1]$ $V_1\ [[0,1,1],\ \ X\ \ ,[1,0,0]]$

$C_m\ [0,2,1]$ $V_m\ [[0,1,1],\ \ X\ \ ,x]$

accept delivery

$C_1 [1,2,1]$ $V_1\ [[0,2,1],\ x,\ [1,0,0]]$