

1. Vector clocks overcome the shortcoming of Lamport logical clocks in the way that they can tell us if  $t(a) < t(b)$  than we know a happened before b. With Lamport clocks we cannot come to this conclusion. We can also detect causal dependency with vector clocks, which we cannot do with Lamport.

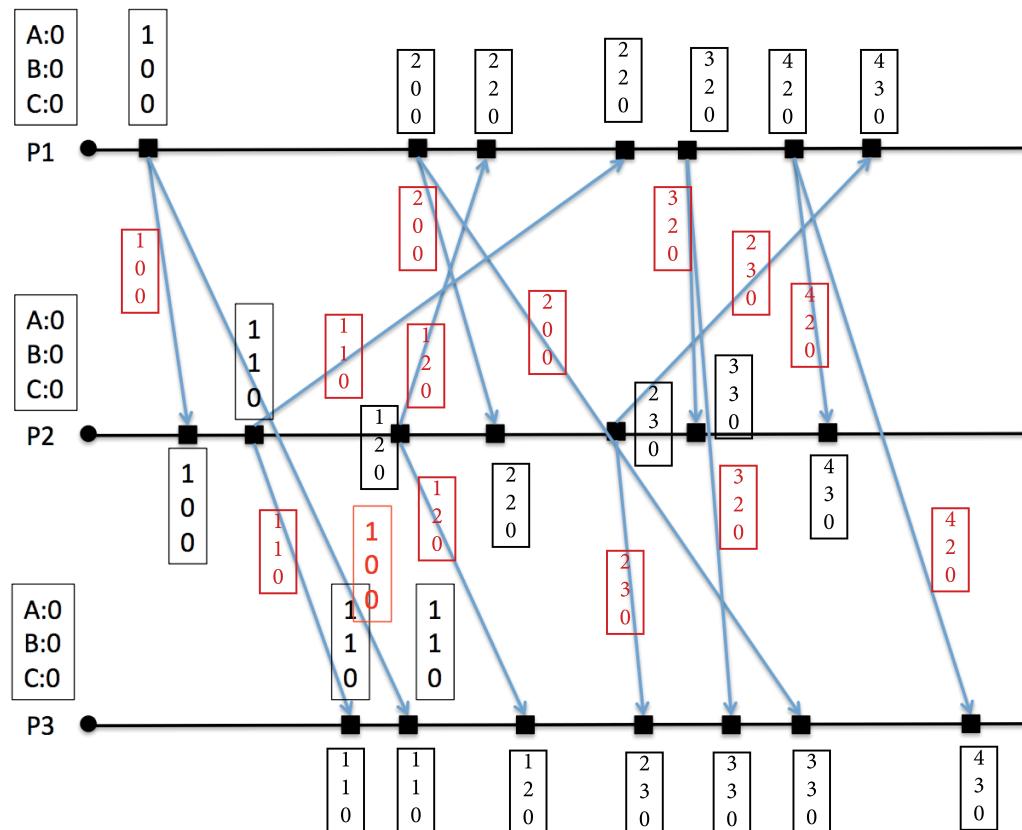
2.  $ta < tb$  if

1. they are not equal timestamps ( there exists  $i$ ,  $ta[i] \neq tb[i]$ ) and
2. each  $ta[i]$  is less than or equal to  $tb[i]$  (for all  $i$ ,  $ta[i] \leq tb[i]$ )

They are concurrent if you cannot compare the two vectors, for example.  $[1, 3, 1]$  to  $[3, 2, 3]$ ...not all  $ta[i]$  are less than or equal to  $tb[i]$ . Therefor they are concurrent and cannot be compared.

3. The clock ticks before sending in order to send the current and updated vector with the correct time stamp. If it ticked after, the vector that was sent would already be out of date.

4.



5. The problem that is solved is the number of processes in the distributed system has to be constant and known in advance.