Design Approach

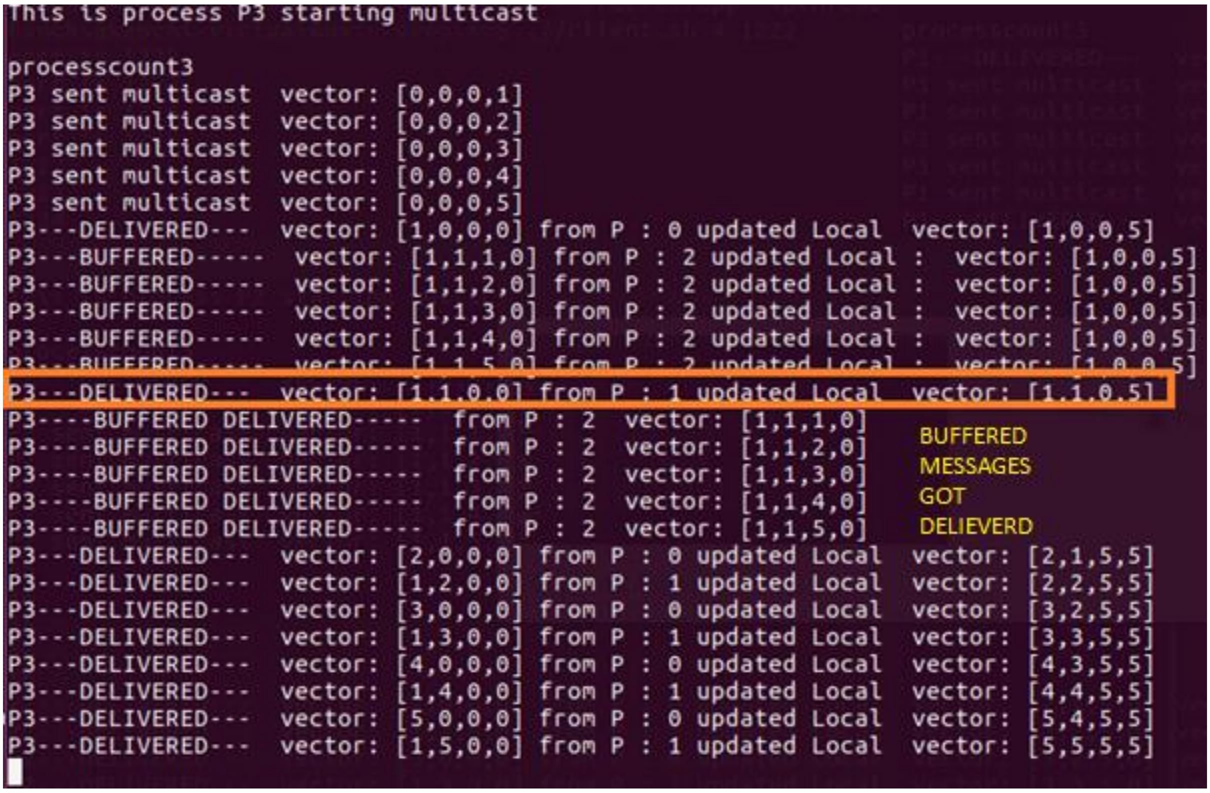
- The processes are spawned with the script file.
- Each process reads the ports number from the shared file and connects to all processes in the distributed system.
- After connecting to other processes, process starts multicasting by sending a multicast message to all other processes.
- In the multicast message, it sends its own vector to other process. Before sending the vector, it increments its vector index by 1 in the vector i.e, P*j*[*j*] = P*j*[*j*] + 1
- At the receiver, If Pi receives a multicast from Pj with vector M[1…N] (= Pj[1…N]) in message, buffer it until both:
    - This message is the next one Pi is expecting from Pj, i.e., M[j] = Pi[j] + 1
    - All multicasts, anywhere in the group, which happened-before M have been received at Pi, i.e., for all k ≠ j: M[k] ≤ Pi[k] i.e., Receiver satisfies causality
- When above two conditions satisfied, deliver M to application and set Pi[j] = M[j]
- The messages are buffered in a vector of structure.
- When a message is delivered to the application, the buffered is checked to find if there is any message which can now be delivered.
- When a message from the buffer could be delivered, the buffer is again iterated to see if any other message can be delivered until no message could be delivered from the buffer.

The below screenshot displays that the messages were buffered until the condition was met **(the highlighted message).** After the condition was met, the buffered messages were delivered in the required order in later execution.

BUFFERED: shows that message is buffered and not delivered to application

DELIVERED: the message is delivered to the application

BUFFERED_DELIVERED: shows the messages which was buffered is now delivered



```
This is process P3 starting multicast

processcount3
P3 sent multicast  vector: [0,0,0,1]
P3 sent multicast  vector: [0,0,0,2]
P3 sent multicast  vector: [0,0,0,3]
P3 sent multicast  vector: [0,0,0,4]
P3 sent multicast  vector: [0,0,0,5]
P3---DELIVERED---  vector: [1,0,0,0] from P : 0 updated Local  vector: [1,0,0,5]
P3---BUFFERED-----  vector: [1,1,1,0] from P : 2 updated Local :  vector: [1,0,0,5]
P3---BUFFERED-----  vector: [1,1,2,0] from P : 2 updated Local :  vector: [1,0,0,5]
P3---BUFFERED-----  vector: [1,1,3,0] from P : 2 updated Local :  vector: [1,0,0,5]
P3---BUFFERED-----  vector: [1,1,4,0] from P : 2 updated Local :  vector: [1,0,0,5]
P3---BUFFERED-----  vector: [1,1,5,0] from P : 2 updated Local :  vector: [1,0,0,5]
P3---DELIVERED---  vector: [1,1,0,0] from P : 1 updated Local  vector: [1,1,0,5]
P3----BUFFERED DELIVERED-----   from P : 2  vector: [1,1,1,0]
P3----BUFFERED DELIVERED-----   from P : 2  vector: [1,1,2,0]
P3----BUFFERED DELIVERED-----   from P : 2  vector: [1,1,3,0]
P3----BUFFERED DELIVERED-----   from P : 2  vector: [1,1,4,0]
P3----BUFFERED DELIVERED-----   from P : 2  vector: [1,1,5,0]
P3---DELIVERED---  vector: [2,0,0,0] from P : 0 updated Local  vector: [2,1,5,5]
P3---DELIVERED---  vector: [1,2,0,0] from P : 1 updated Local  vector: [2,2,5,5]
P3---DELIVERED---  vector: [3,0,0,0] from P : 0 updated Local  vector: [3,2,5,5]
P3---DELIVERED---  vector: [1,3,0,0] from P : 1 updated Local  vector: [3,3,5,5]
P3---DELIVERED---  vector: [4,0,0,0] from P : 0 updated Local  vector: [4,3,5,5]
P3---DELIVERED---  vector: [1,4,0,0] from P : 1 updated Local  vector: [4,4,5,5]
P3---DELIVERED---  vector: [5,0,0,0] from P : 0 updated Local  vector: [5,4,5,5]
P3---DELIVERED---  vector: [1,5,0,0] from P : 1 updated Local  vector: [5,5,5,5]
```

BUFFERED
MESSAGES
GOT
DELIEVERD