Unit - 2

1) Message ordering paradigms:

*Message delivery in orderly manner is very important because it determins the behaviour of messaging.

*Message ordering paradigms are classified into four different types they are:

* non-fifo
* FIFO
* Causal order
* Synchronous order

*All Physical Links obey FIFO rule.

*Logical links are inherently non-FIFO. It can assume connection oriented service at transport layer

for example: TCP.

* To implement FIFO over non-FIFO link, use number and connection-id for each message.

* FIFO execution is an A-execution in which for all (s, r) and $(s', r') \in T$ $(s \sim s'$ and $r \sim r'$ and $s < s') \Rightarrow r < r'$.

* Causally order execution is an A-execution in which for all (s, r) and $(s', r') \in T$, $(r \sim r'$ and $s < s') \Rightarrow r < r'$

* There are no causality cycles on any logical link.

* Synchronous order is used for pair communication between the processes.

* Both send and receive message event appears instantaneous.

*Causal Order:

*Causal ordering of messages was proposed by Birman and Joseph.

*This message ordering paradig is similar to many to many communication

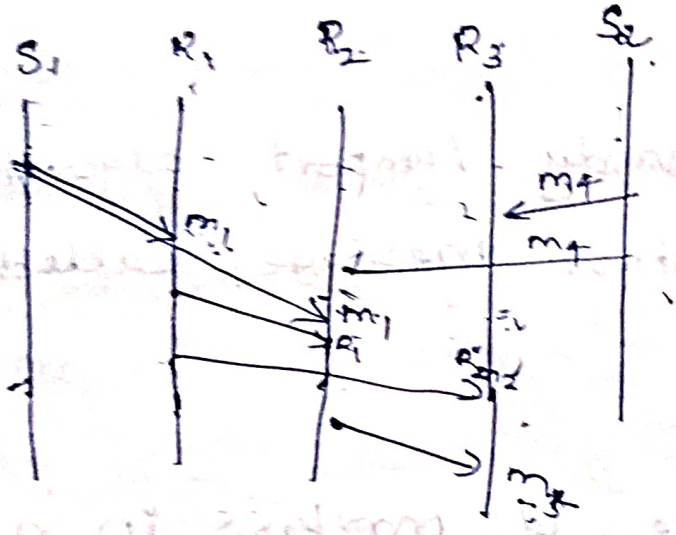
*Multiple sender sends message to multiple receivers. in Ordered message delivery.

*The purpose of causal ordering of messages in ensure the same causal relationship between message send and message receive events.

*If Send(m_1) happens before Send(m_2), then both messages m_1 and m_2 must receive m_1 before m_2.

* If two message sending events are not causally related, then the message receives in any order.

* Two message sending events are said to be causally related if they are correlated.



2) Snapshot algorithms for FIFO channel.

A Snapshot captures the local states of each process along with the state of each communication channel.

Snapshots are required to

1) checkpointing

2) collecting garbage

3) Detecting deadlocks.

4) Debugging.

Chandy - Lamport algorithm:

* This algorithm will record a global snapshot for each process chanel.

* The Chandy - Lamport algorithm uses a control message, called marker.

* A role of markers in a fifo system is to act as delimiters for the messages in the channels.

* If a site is recorded its snapshots, then immediately sends a marker to channels.

marker Sending rule for Process Pi

Step 1: Process Pi records it's State

Step 2: Pi Sends a marker to Channel C

marker receiving rule for process Pi

* if Pj has not recorded it's State then Execute "marker Sending rule".

* Else

Record the State of C as a Set of messages.

Initiating a Snapshot:

* process Pi Initiates the Snapshot

* Pi records its own State and Send the marker message to all other processes

Propagating a Snapshot:

* for all processes Pj Consider a message on channel Ckj

* If marker message is seen for the first time

Then Pj records own state and marks Ckj as empty

* Else add all the messages to channels

Termination of a Snapshot:

* If All processes have received a marker. Then terminate the process.

* All process. have received a marker on all the N-1 incoming Channel.

Complexity:

O(e) Messages

O(d) Times.

e - refers edges of Network

d - diameter of Network

Assumptions:

* No failure.
* All message arrive intact, exactly once
* Communication channels are unidirectional and FIFO-ordered.
* Any process may initiate the Snapshot
* Snapshot does not interfere with normal execution.

Group Communication.

Group communication can be implemented by three ways they are

* one to many
* Many to one
* Many to Many

Group Management:

Here receivers froms a group. Group is of two types.

* Closed group:- only group members can send message to outside the process.
* Open group: Any process in the system can send Message

Group addressing:

Group addressing is classified into three types they are.

* High level
* Low level
* User application

* High level:

ASCII string

* Low level:

Underlying hardware

* User application

programming languages.