Chapter - 3
Interprocess Communication

Message Passing

Message Passing vs Shared Memory

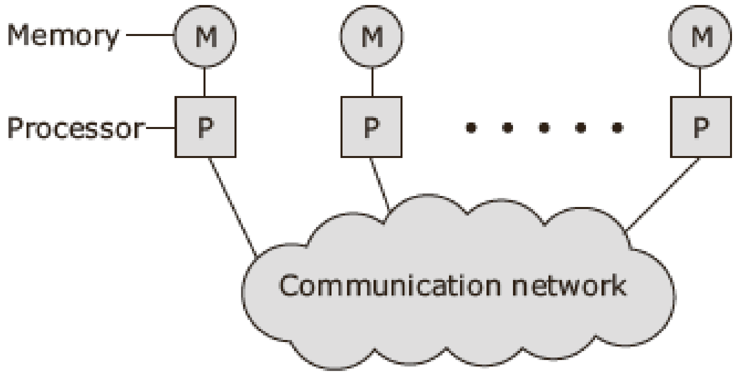


Figure 3-1 Typical message-passing operation

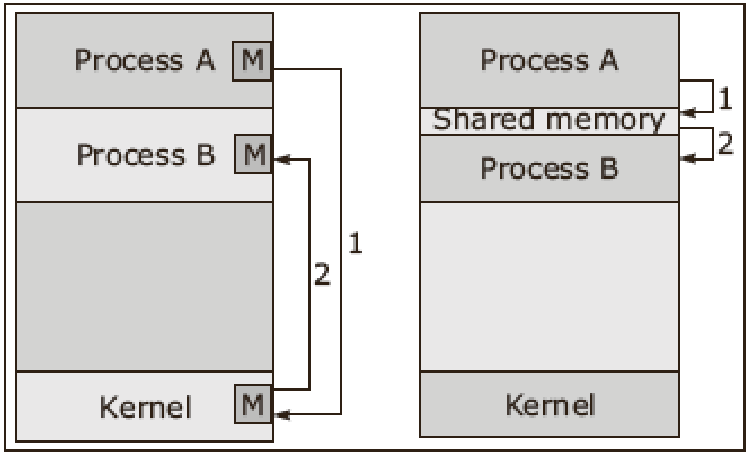


Figure 3-2 Message-passing vs. Shared memory approach

Desirable Features of Message Passing Systems

- Hardware approach
- Functionality
- Performance
- Uniform semantics
- Efficiency
- Reliability
- Correctness
- Flexibility
- Portability
- Security

Message passing process

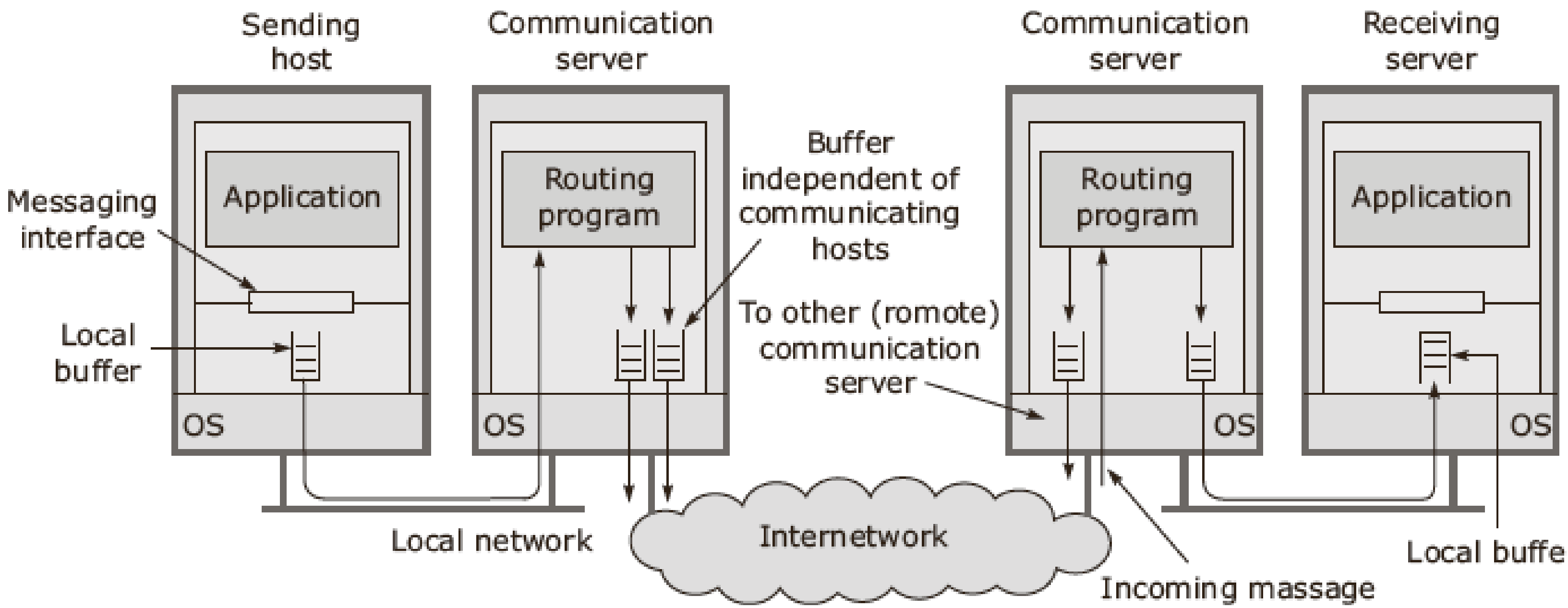


Figure 3-3 Message passing operation

IPC message format

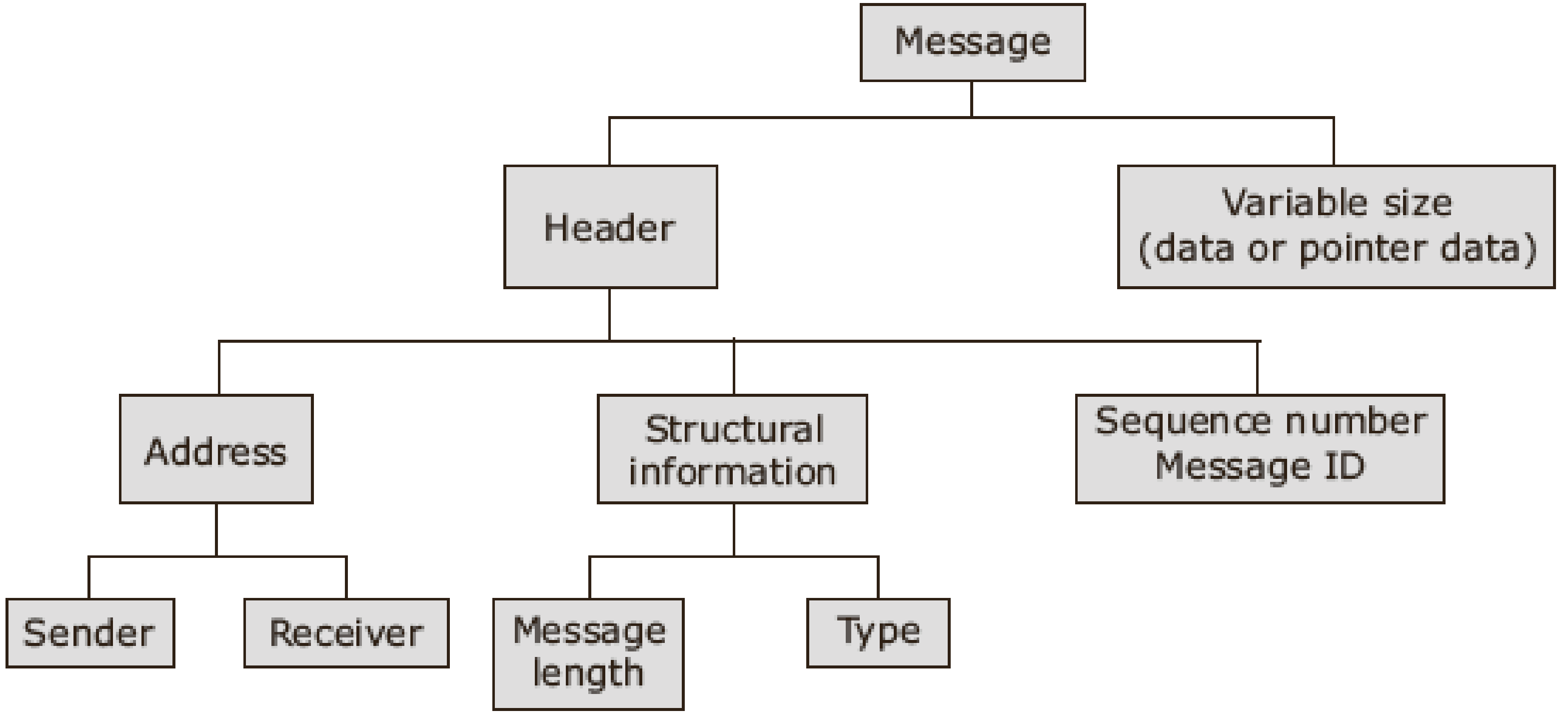
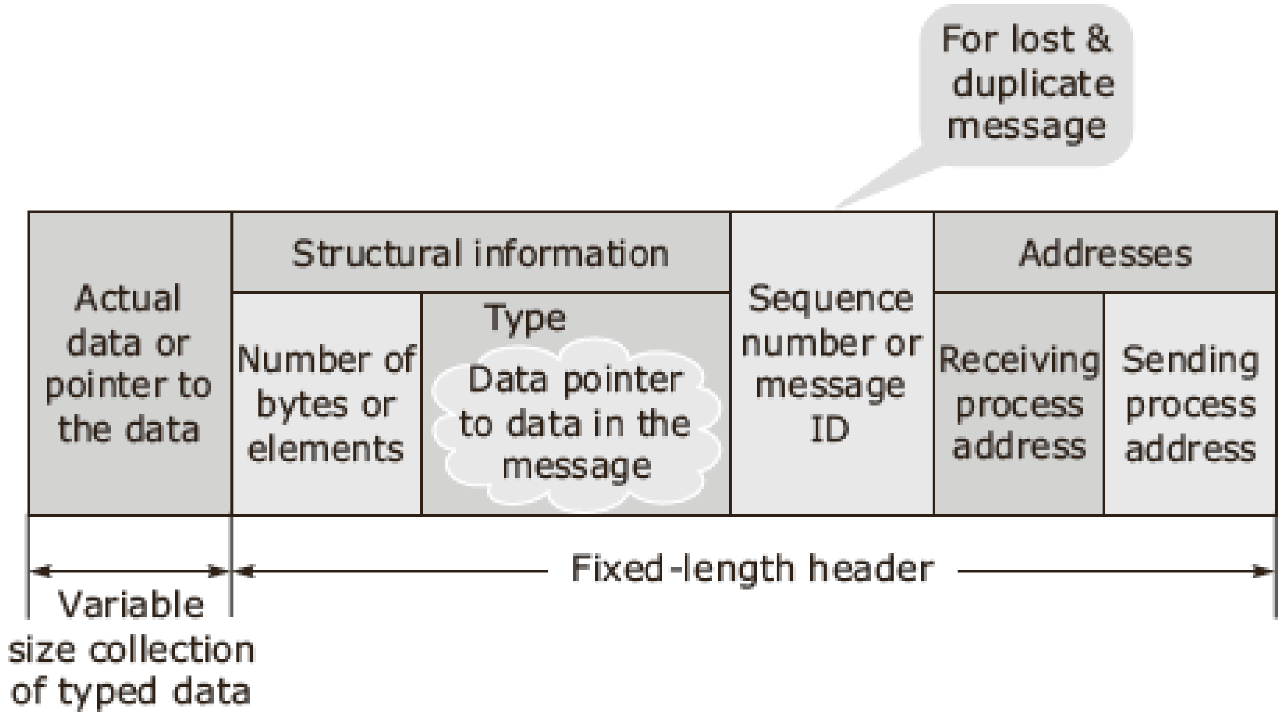


Figure 3-4 Components of an IPC message

IPC Message



Sender and receiver are fully aware of the message formats used in the communication process and the mechanisms used to send and receive messages.

Figure 3-5 A typical IPC message structure

IPC synchronization

- Ensure message is received in the buffer:
  - Polling
  - Interrupt
- Message communication techniques
  - Synchronous communication
  - Asynchronous communication

IPC: Synchronous communication

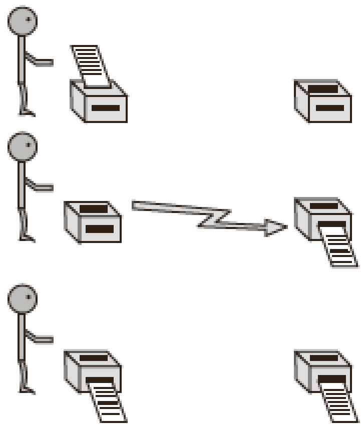


Figure 3-6 Synchronous communication operation

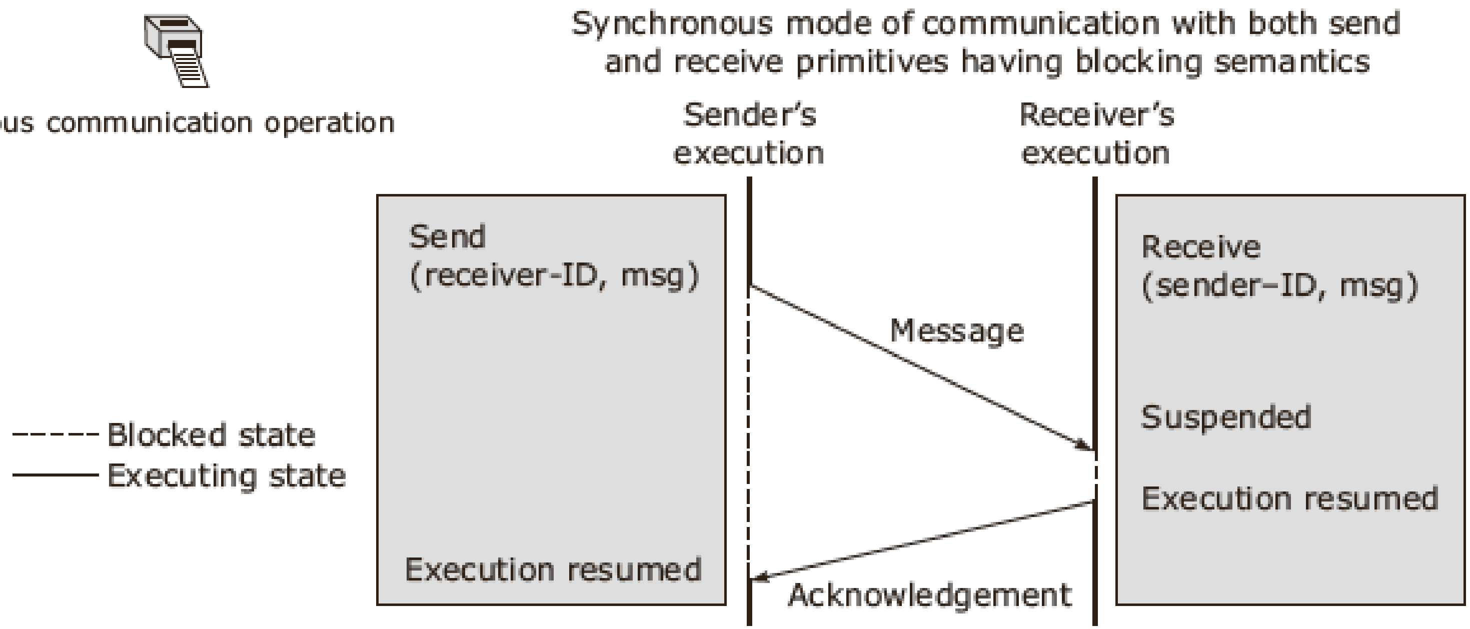


Figure 3-7 Synchronous communication operation

IPC: Asynchronous communication

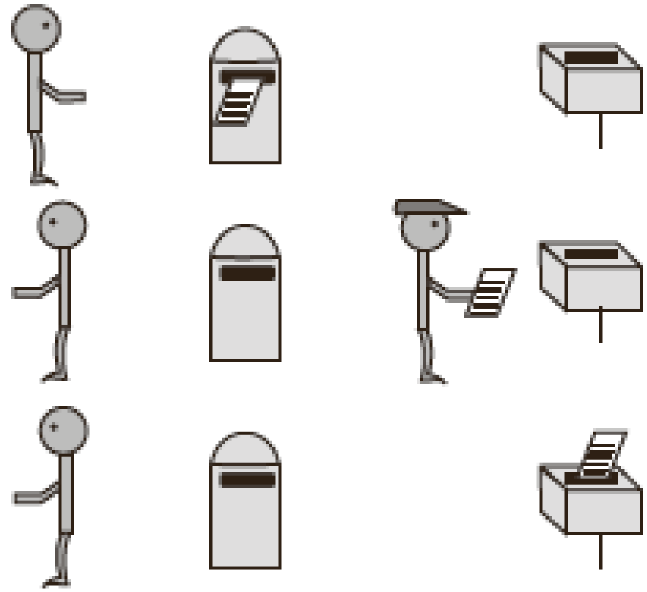


Figure 3-8 Asynchronous communication operation

IPC primitives

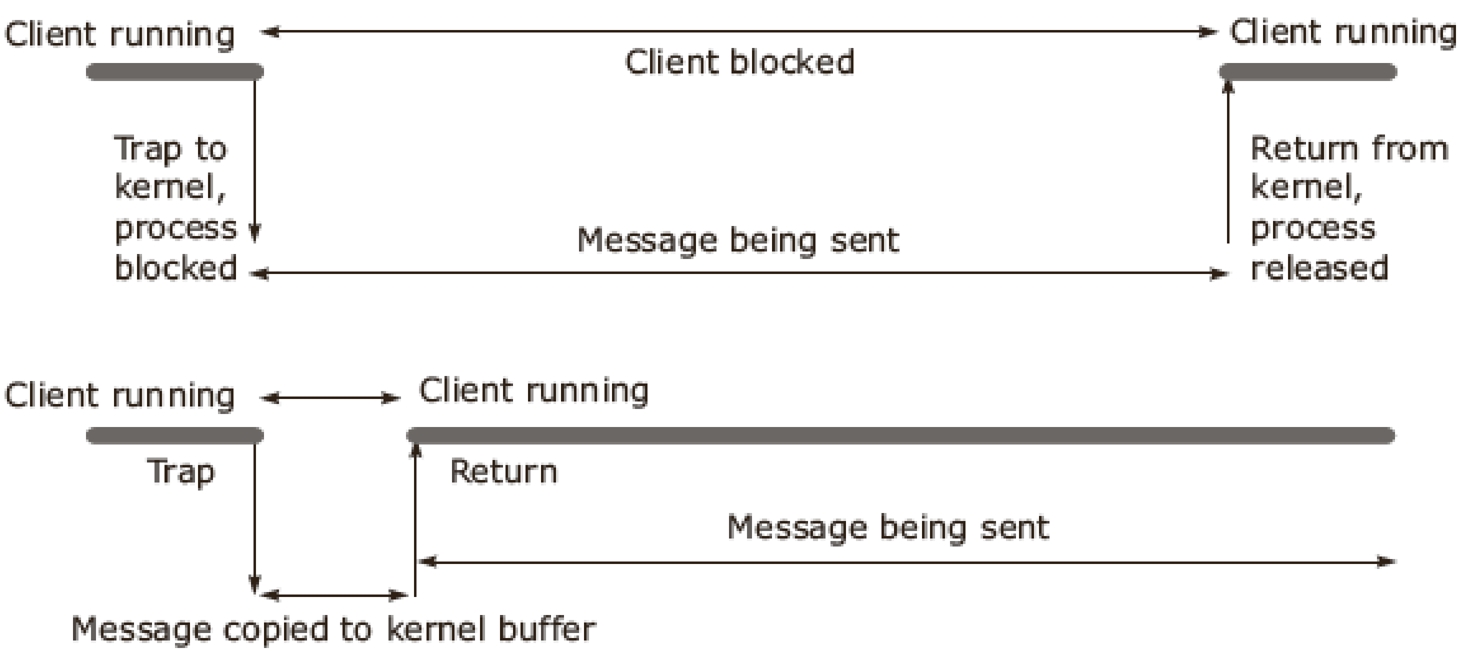


Figure 3-9 Blocking and non-blocking primitives

Message buffering strategies

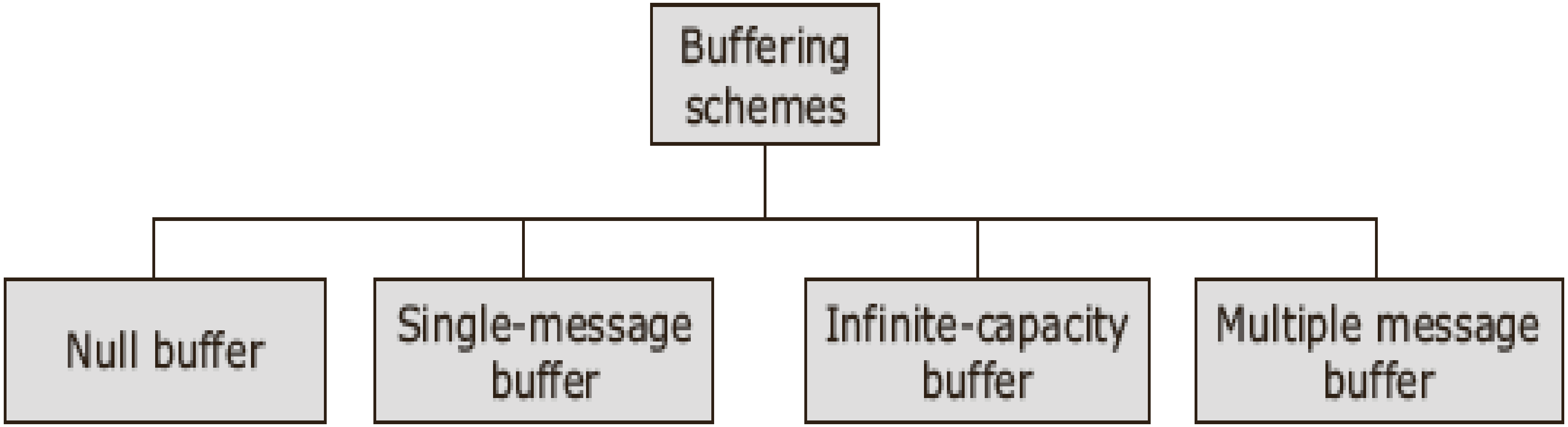


Figure 3-10 Buffering schemes

Null buffering

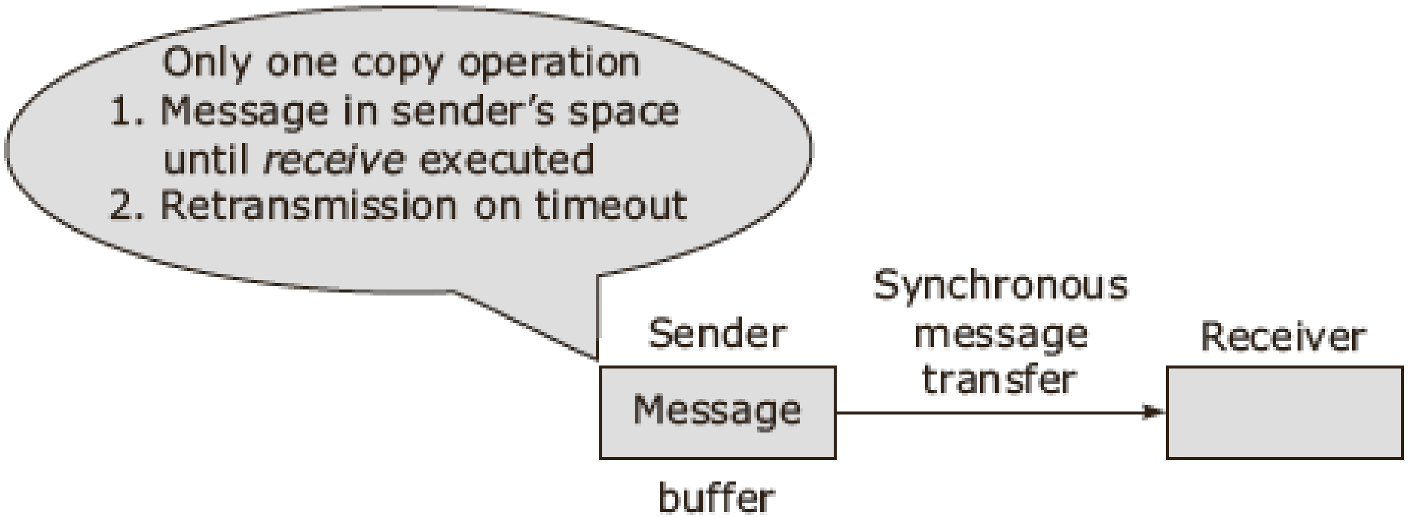


Figure 3-11 Message transfer with no buffering and single copy operation

Null buffering with blocked receiver

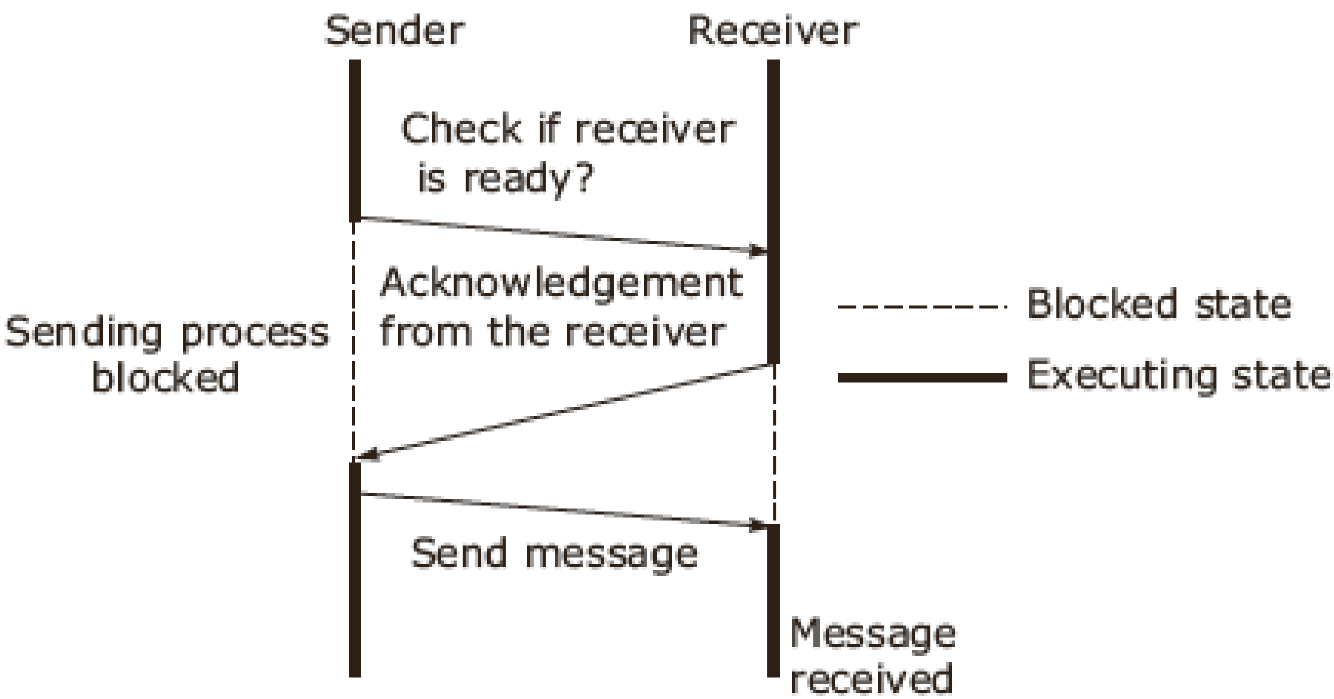


Figure 3-12 Null buffering with effective message-passing blocking mechanism

Null buffering with non blocked receiver

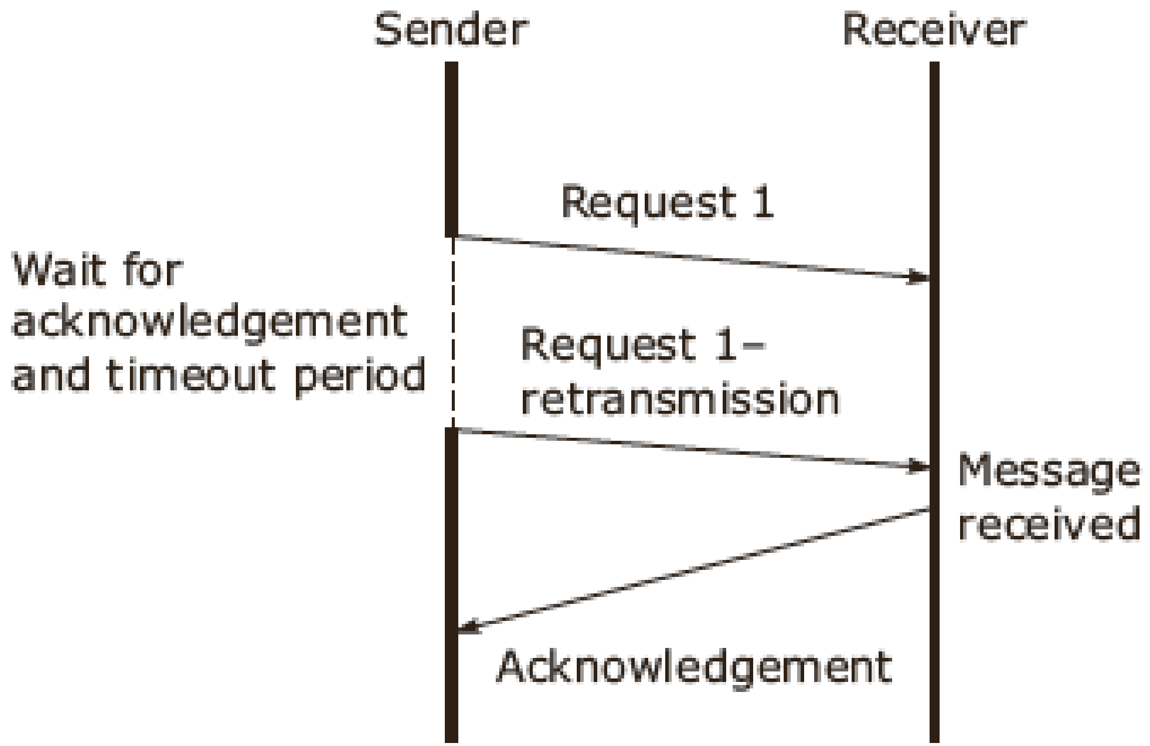


Figure 3-13 Null buffering with effective message-passing

Message buffering: single buffer

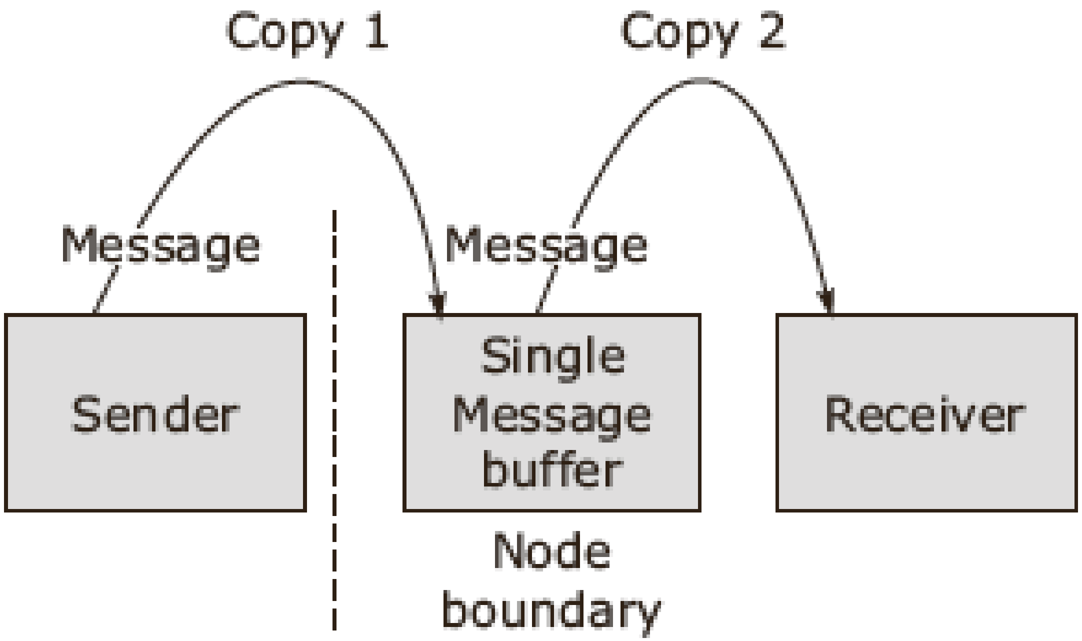
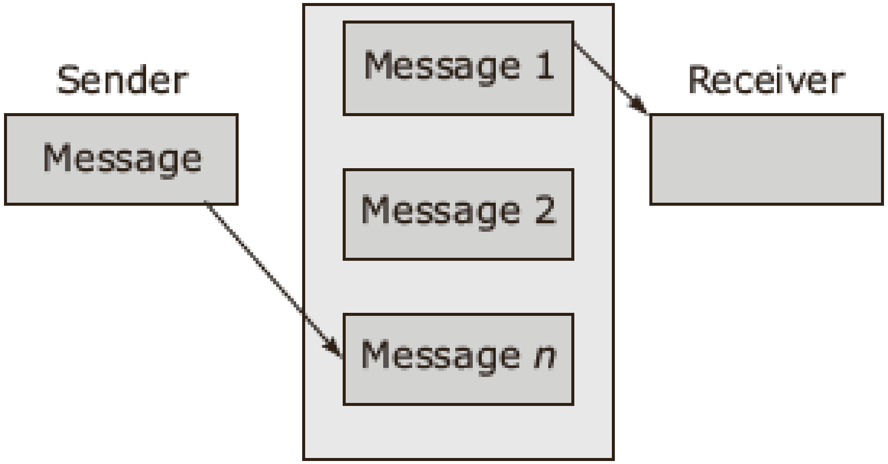


Figure 3-14 Single-message buffering

Message buffering: multiple message buffer

- Receiver overflow handled using:
  - Unsuccessful communication indication
  - Flow control mechanism



Asynchronous message transfer

Figure 3-15 Multiple-message buffering

Multidatagram messaging

- Concept of MTU
- Message sequencing and reassembly
- Message contents
- Message representation: tagged, untagged

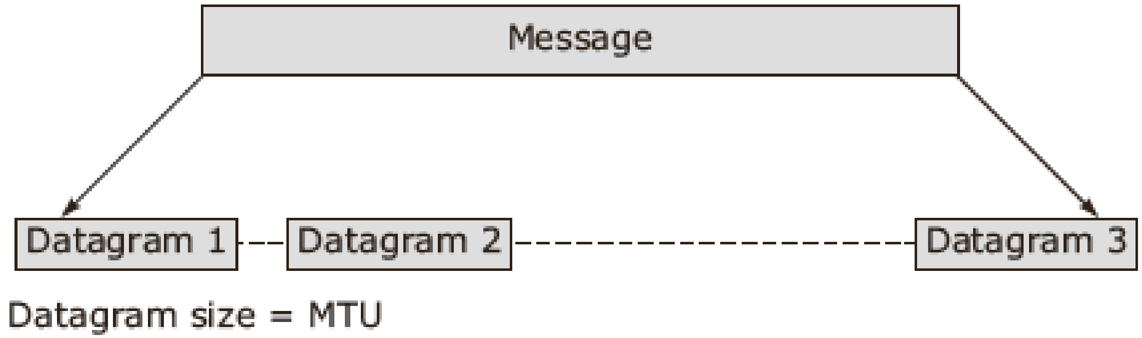


Figure 3-16 Multidatagram message

Message data transmission

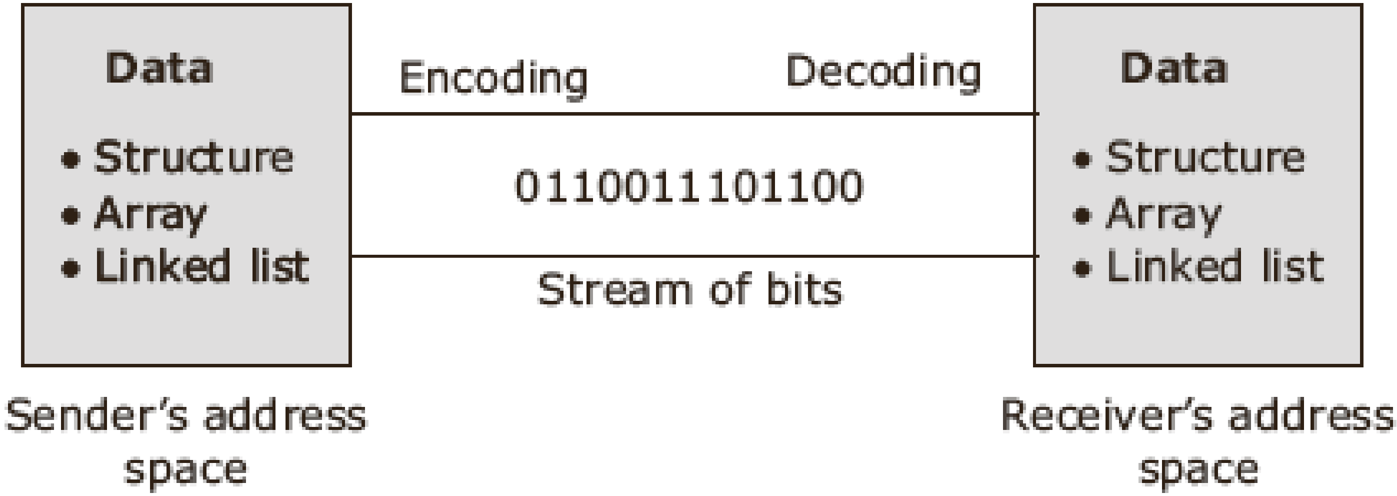


Figure 3-17 Encoding/decoding message data

Process addressing techniques

- Explicit addressing :*send (process_id, message)* and *receive (process_id, message)*
- Implicit addressing:*send_any (service_id, message)* and *receive_any (service_id, message)*
  - Two level addressing : **machine_id@local_id** (receiver machine name)
  - Three level addressing: **machine_id@local_id@machine_id**. (Node where the process was created @ generated by the first machine@ last known location of the machine)

Link-based Process Addressing

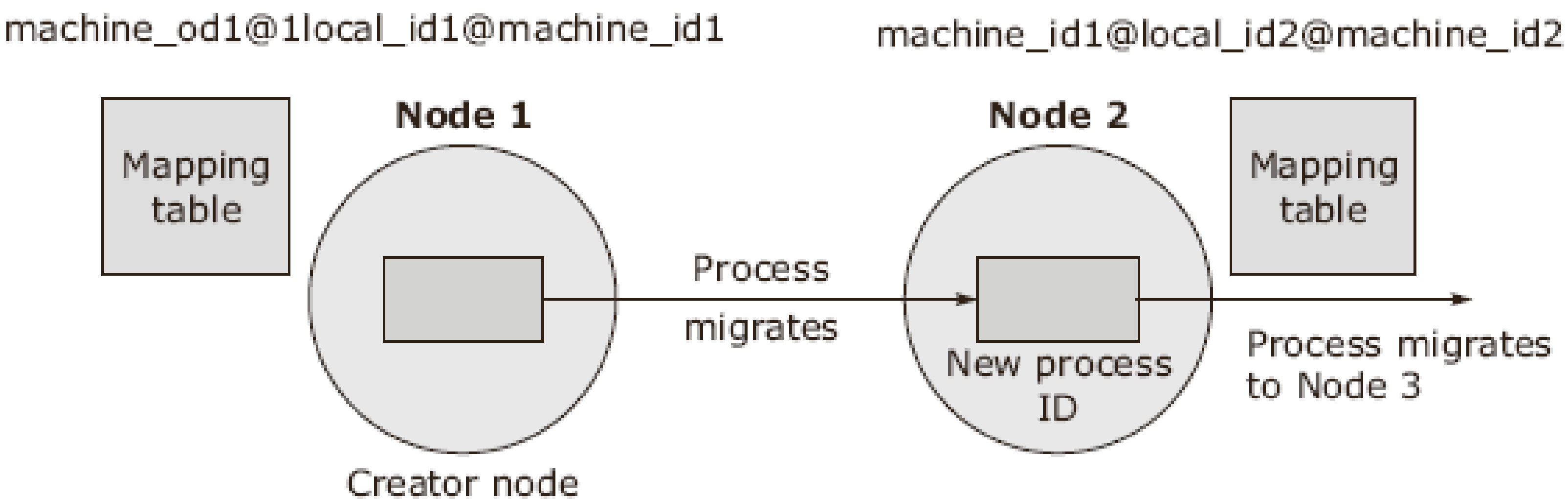


Figure 3-18 Link-based process addressing

Failure handling mechanism

- IPC problems due to system failures

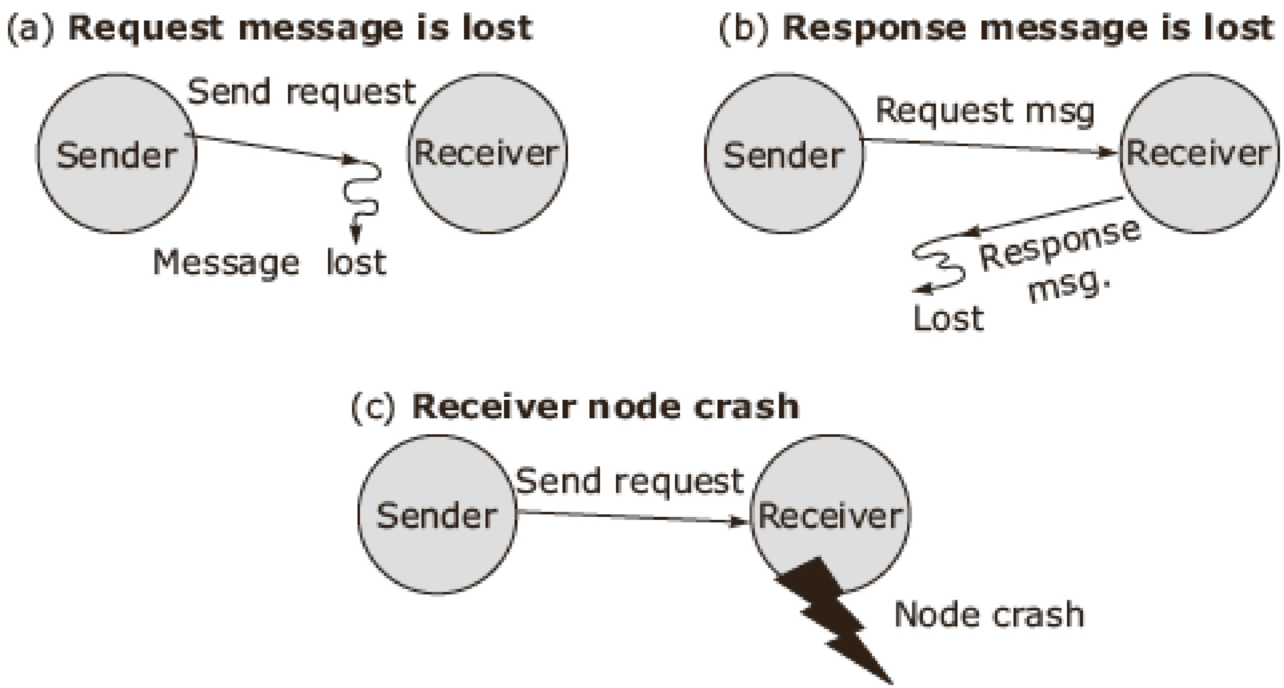


Figure 3.19 IPC problems due to system failures

IPC Protocols

- 4-message reliable IPC protocol
- 3-message reliable IPC protocol
- 2-message reliable IPC protocol

IPC 4 message protocol

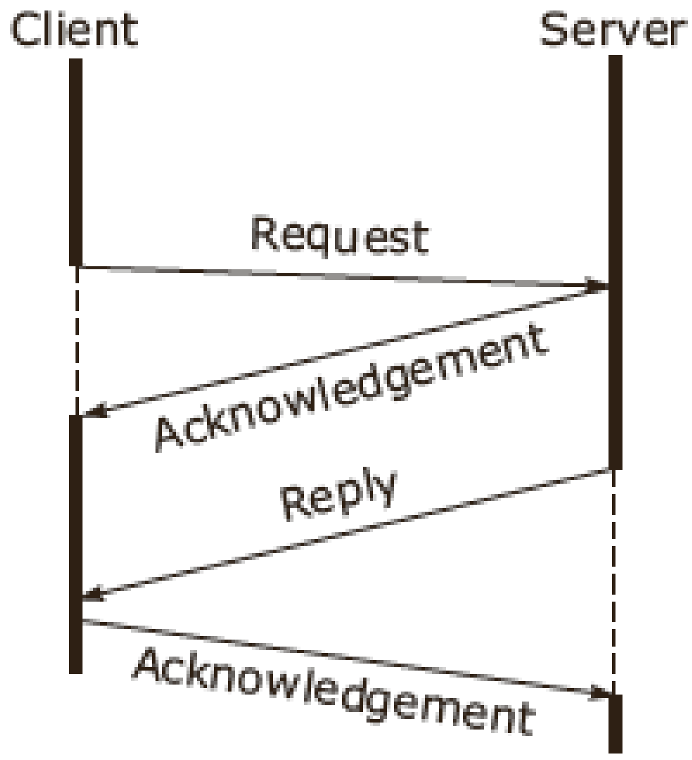


Figure 3-20 4-message reliable IPC protocol

IPC 3 message protocol

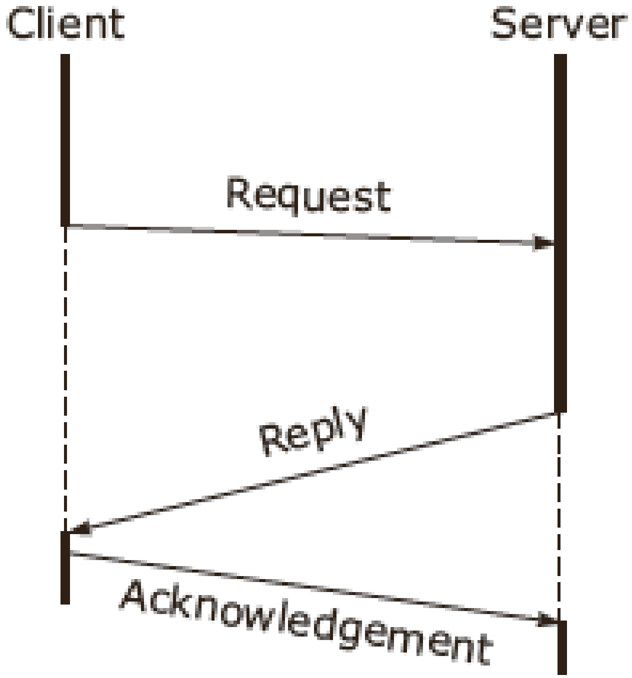


Figure 3-21 3-message IPC protocol

IPC 2 message protocol

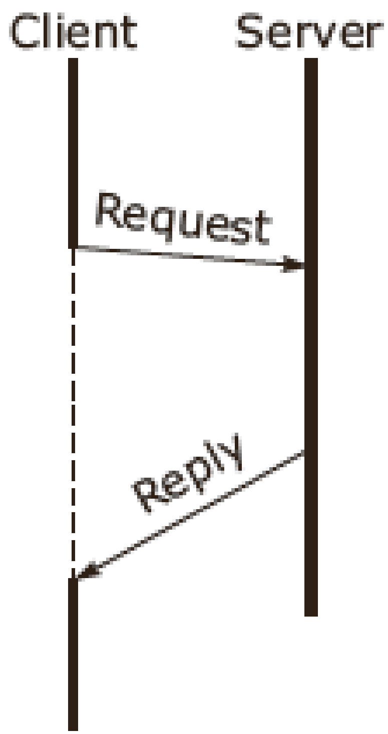


Figure 3-22 2-message IPC protocol

IPC Failure

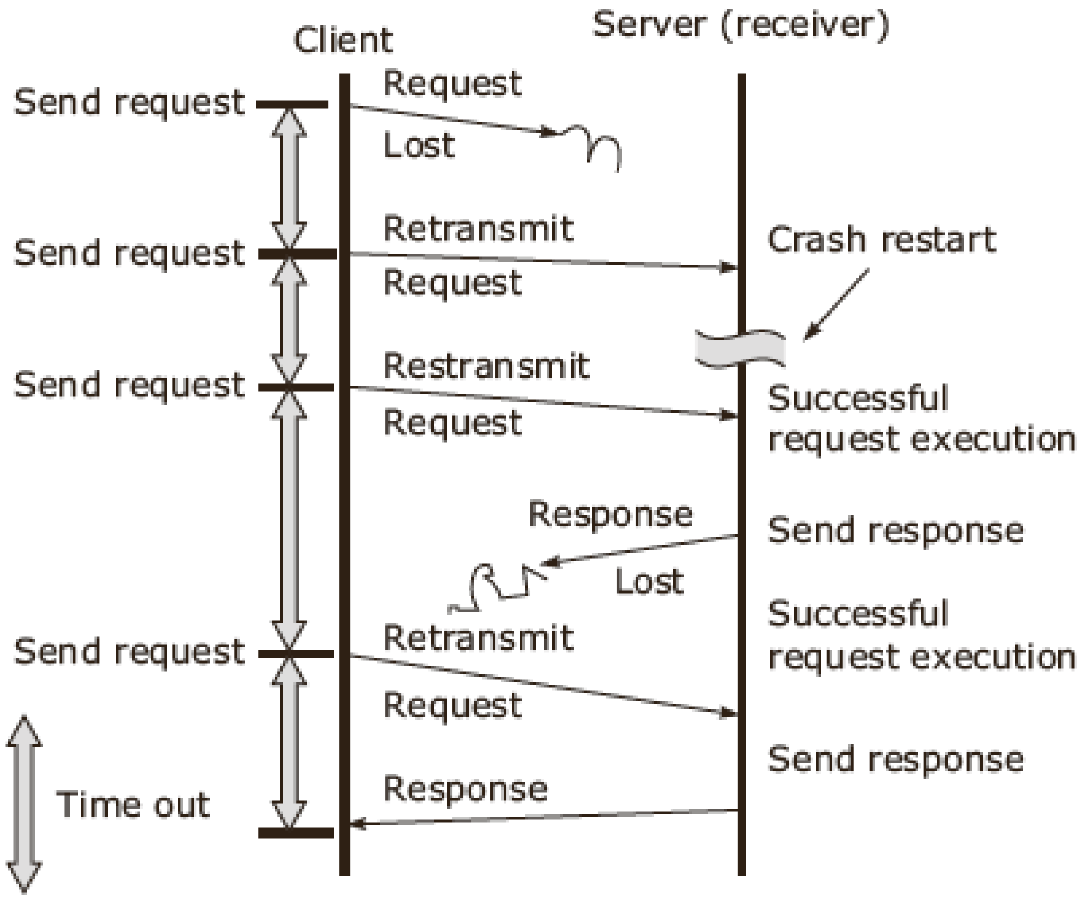


Figure 3-23 Failure handling mechanism

Case Study: IPC in MACH

Case Study: IPC in MACH

- MACH IPC Components
  - Ports
  - Messages
- Message format
- NetMsgServer

MACH message format

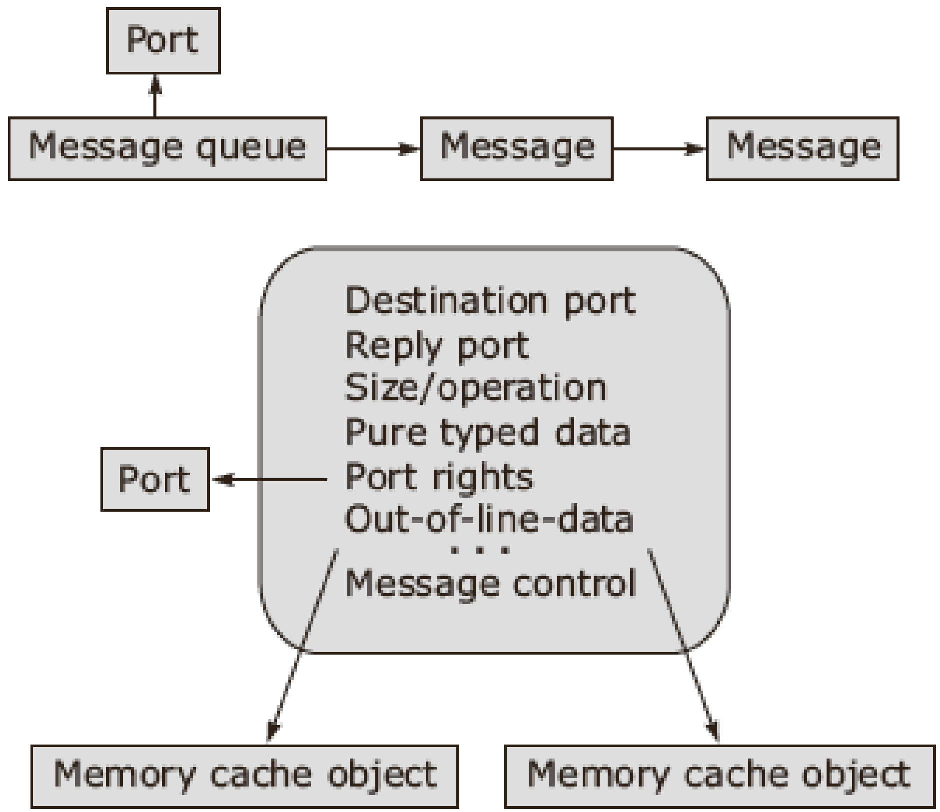


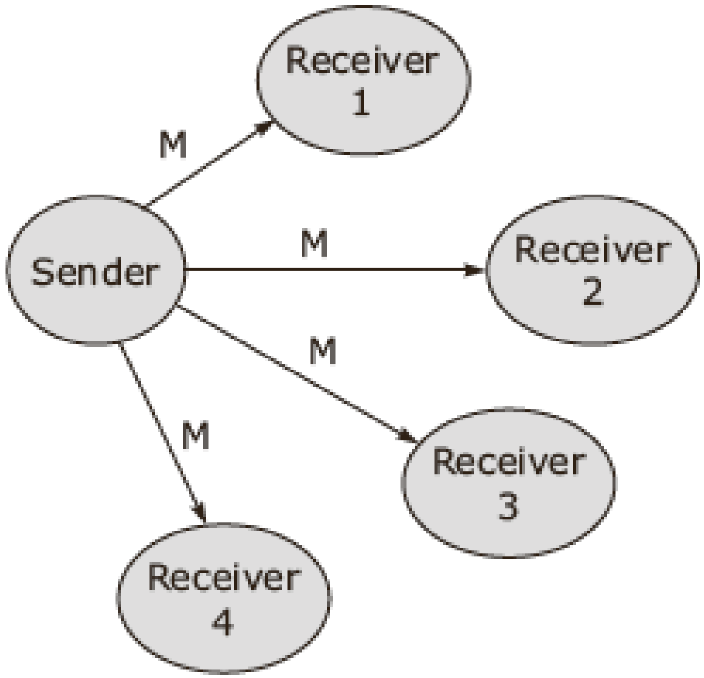
Figure 3-24 Mach message

Group communication

Group Communication

- Unicast –one to one communication
- Many-to-one group communication
- One-to-many or multicast group communication

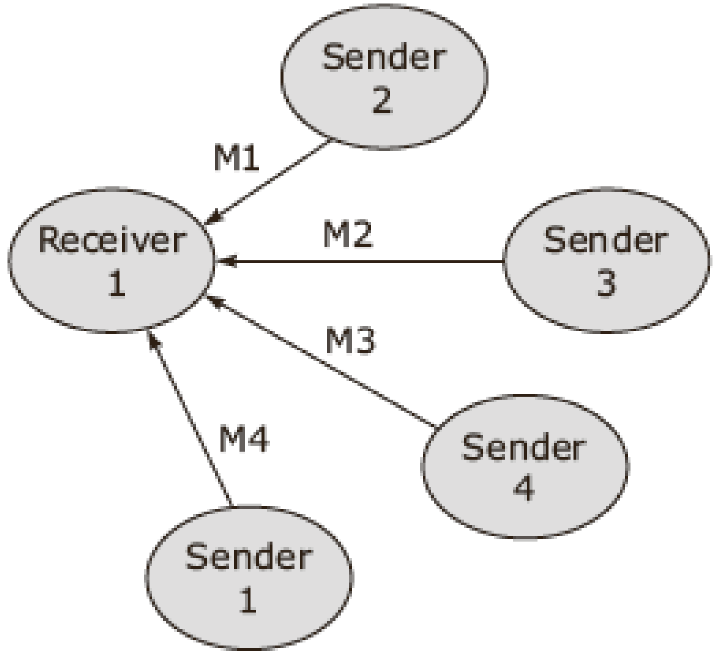
Unicast group communication



Same massage M sent to all nodes

Figure 3-25 Unicast communication

Many to one communication



Multiple senders send message to the same receiver

Figure 3-26 Many-to-one communication

Multi cast communication

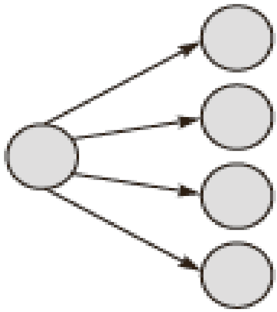


Figure 3.27 One-to-many communication

Broadcast communication

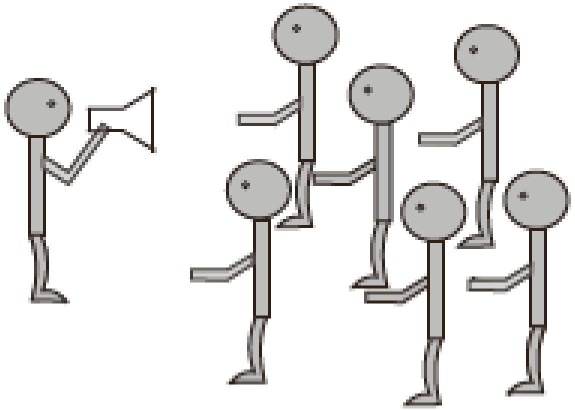


Figure 3-28 A broadcast sends a message to all the recipients

Types of Groups

- Closed group
- Open group
- Peer group
- Hierarchical group

Group management

- Centralized approach
- Distributed approach

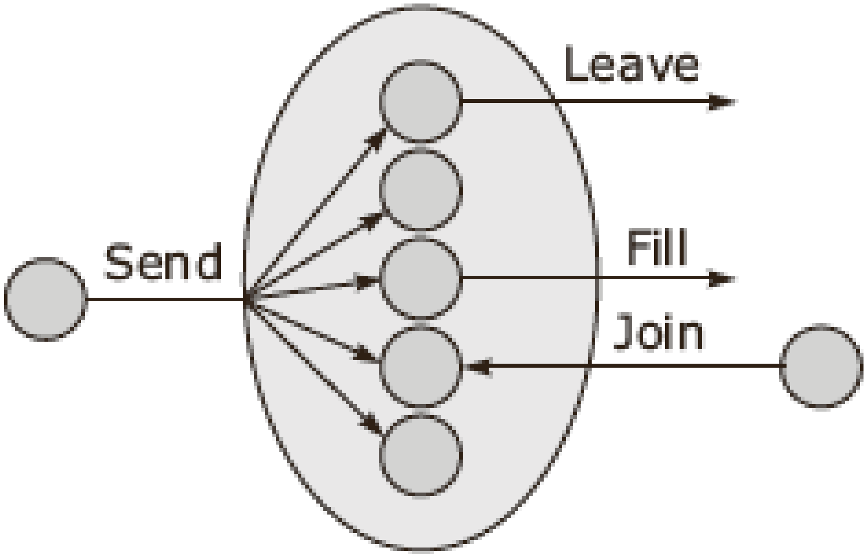


Figure 3-29 Group dynamics

Group addressing

- High level naming
- For large LANs/ MANs: send message to individual group members

Message delivery

- Send to all semantics
- Bulletin board semantics

Reliability mechanism

- Classified based on number of receivers from which sender expects a response

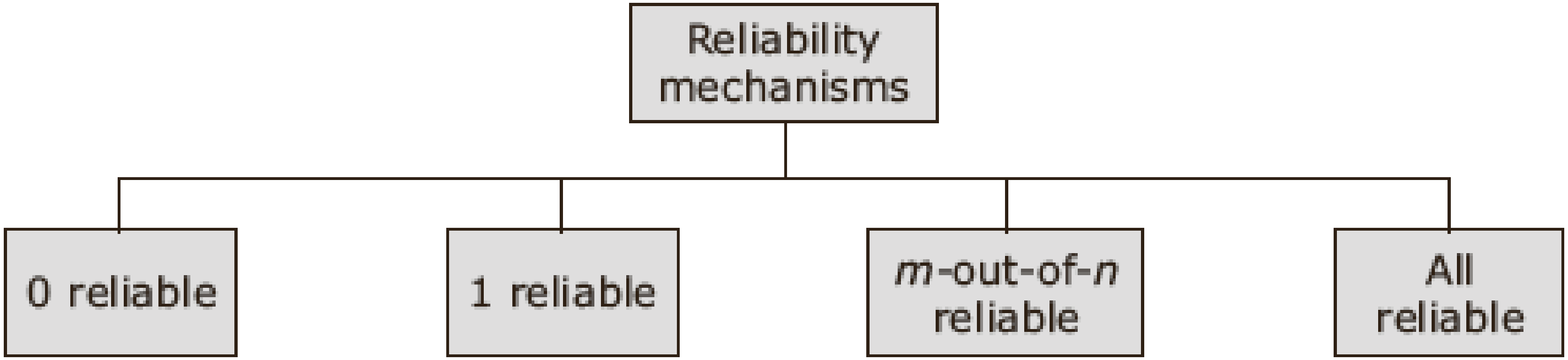


Figure 3-30 Reliability mechanisms

Message ordering

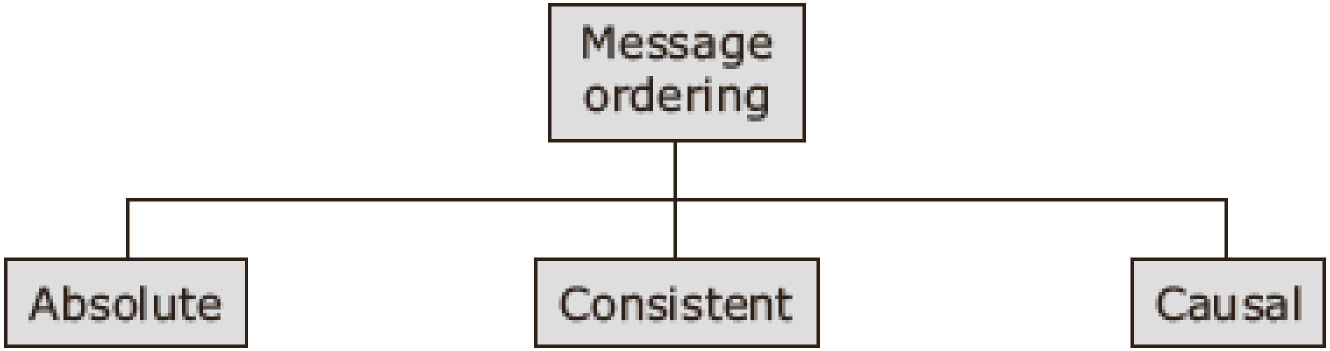


Figure 3-31 Types of message ordering

Message ordering: Absolute Ordering

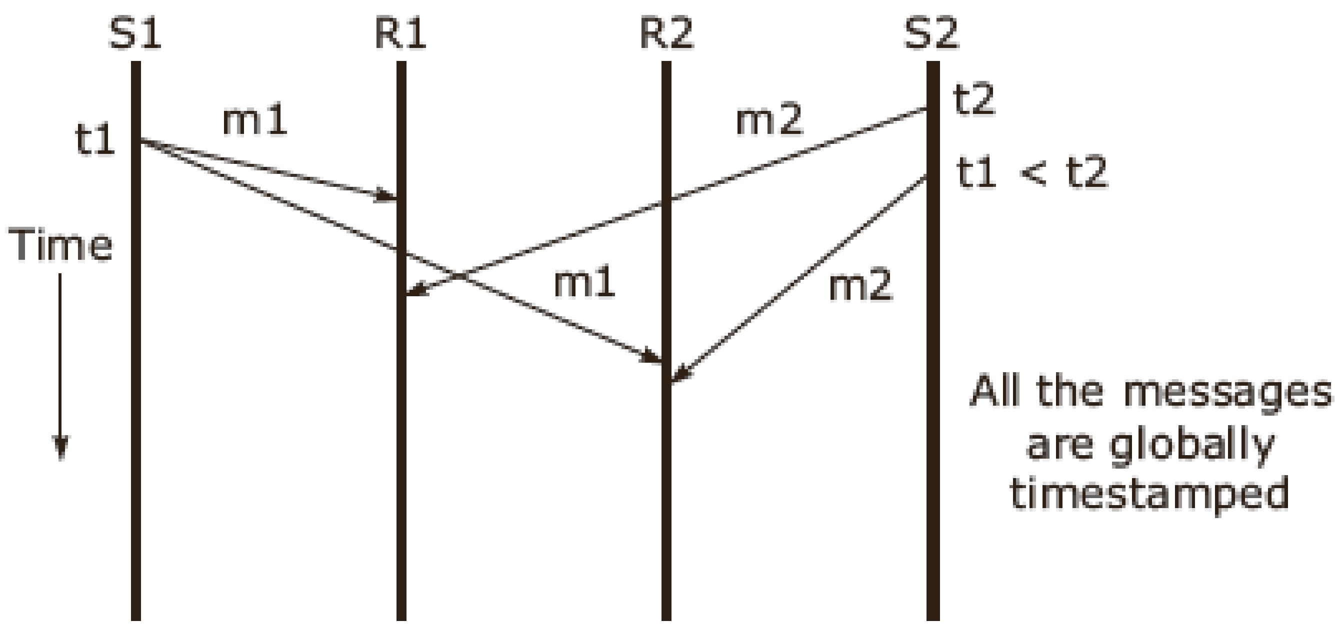


Figure 3-32 Absolute ordering

Message ordering: Consistent ordering

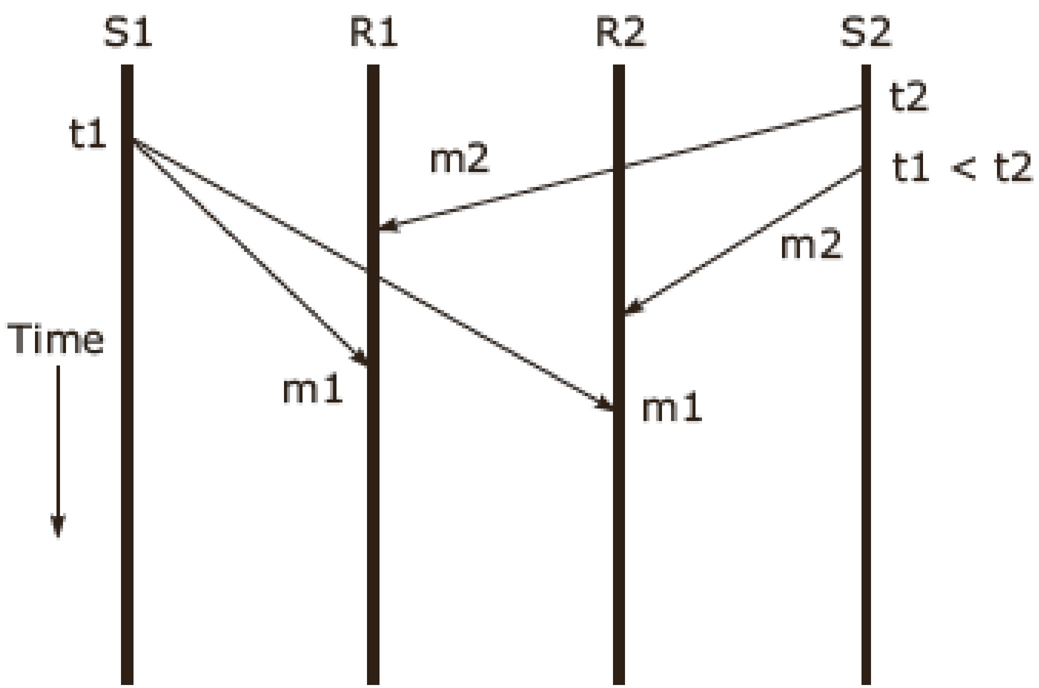


Figure 3-33 Consistent ordering

Message ordering: Causal ordering

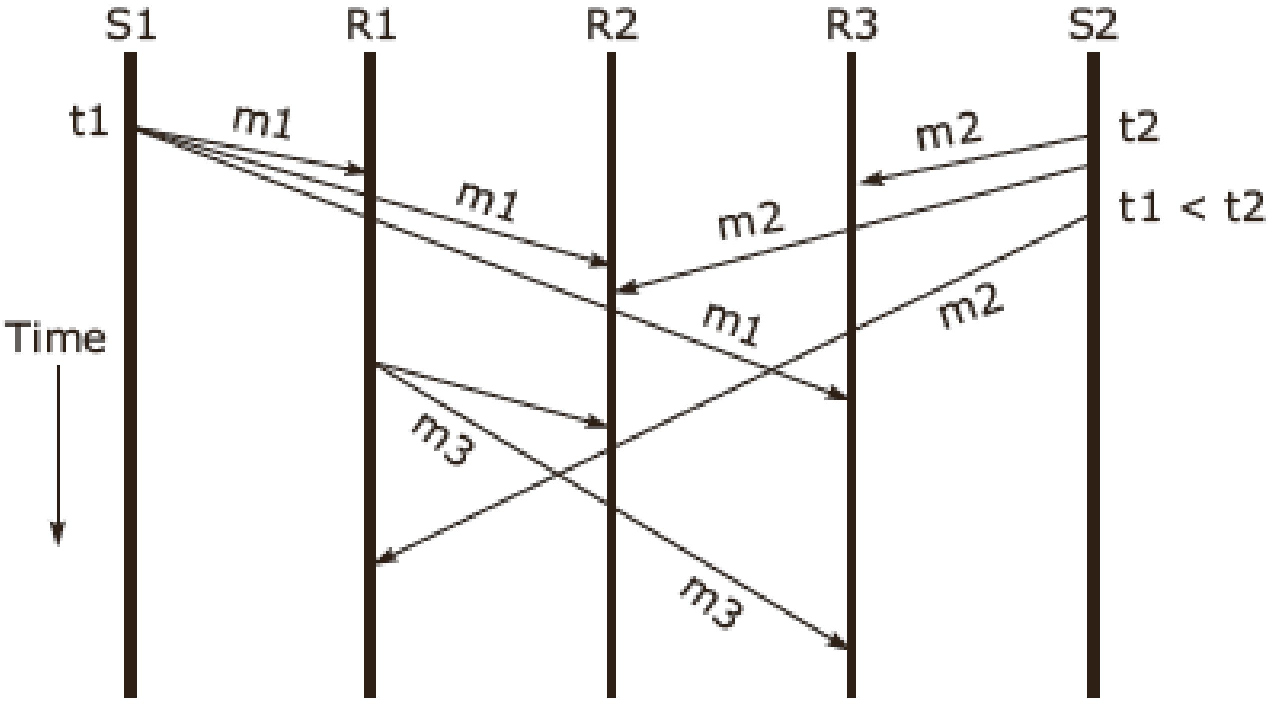


Figure 3-34 Causal ordering

Case Study: CBCAST protocol in ISIS

CBCAST protocol

- S: vector of the sending process attached to the message
- R : vector of the receiving process
- i: sequence number of the sender process
- Runtime system tests following conditions
  - $S[i] = R[i] + 1$
  - $S[j] <= R[j]$ for $j <> i$

CBCAST protocol in ISIS

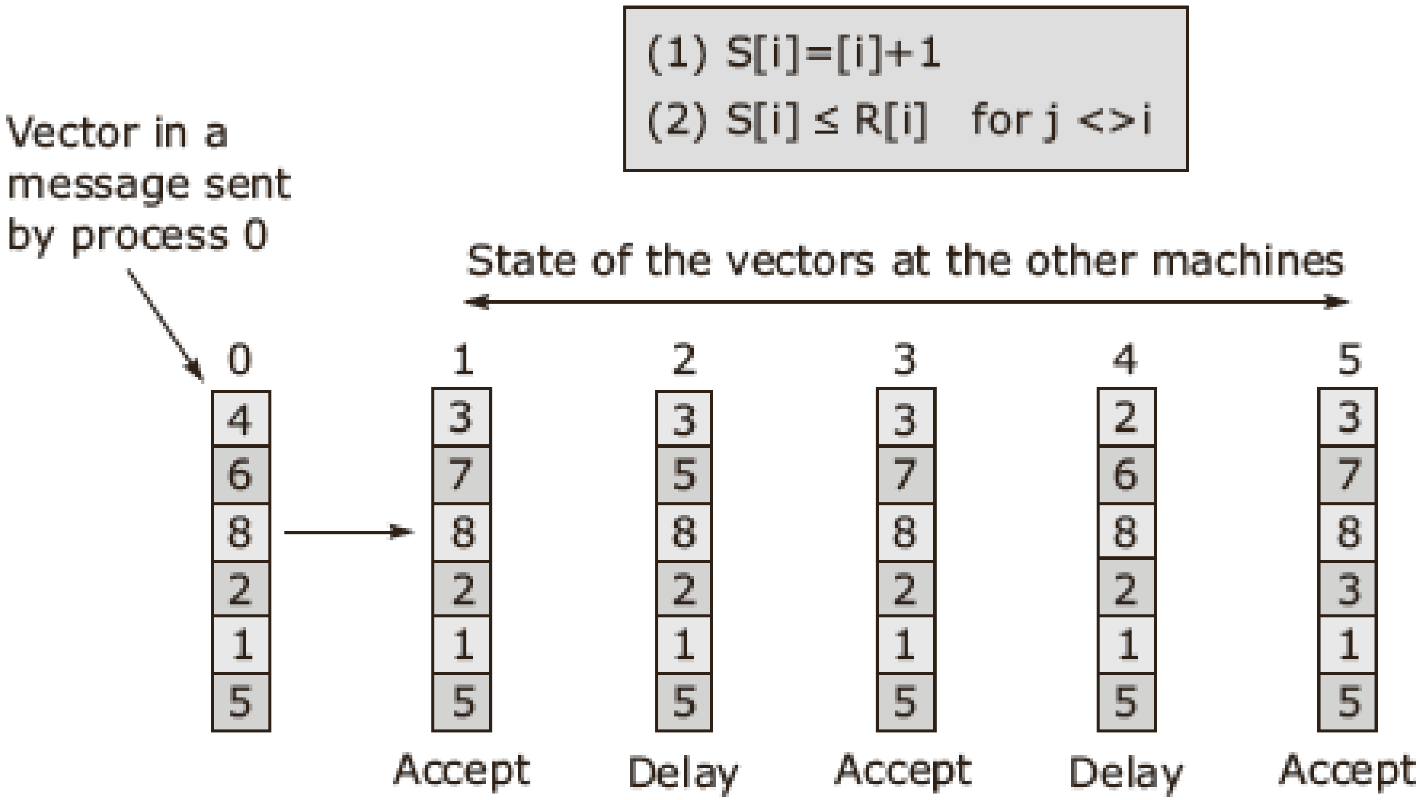


Figure 3-35 CBCAST in ISIS

Summary

- Message Passing
- Case Study: IPC in MACH
- Group communication
- Case Study: CBCAST protocol in ISIS