

Ch-6

Indirect communication

Indirect communication is defined as communication b/w entities of an distributed system through an intermediate elements with no direct coupling b/w the sender and the receiver. It refers 2 key properties :-

- (i) Space uncoupling
- (ii) Time uncoupling

Space uncoupling - In this sender does not know or need to know the identity of the receiver and vice versa.

Time uncoupling - In this sender and receiver can have independent ^{lifetime} time that means sender and receiver do not need to exist at the same time to communicate

Note

1. Explain space coupling in ds with an example.

Time-coupled

Space coupling

Properties: Communication directed towards a given receiver or receivers; receivers must exist at that moment in time.

Eg - Message passing, remote invocation.

Time-uncoupled

properties: Communication directed towards a given receiver or receivers; senders and receivers can have independent lifetimes.

Eg - Message passing, IP multicast.

Time-coupled

space uncoupling

properties:- Sender does not need to know the identity of the receivers; receivers must exist at the moment in time.

Eg - IP Multicast

Time-uncoupled

properties:- Sender does not need to know the identity of the receivers; Sender and receiver can have independent lifetimes

Eg - Most indirect communication paradigms

Group Communication

It is an example of indirect communication. It offers a service whereby a message is sent to a group and then this message is delivered to all members of the group. In this process the sender is not aware of the identities of the receiver.

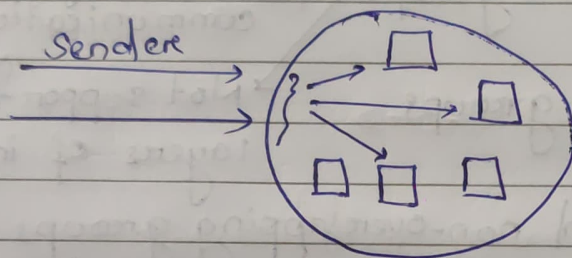
following are the key applications of group communication:-

- the reliable discrimination of information to potentially large numbers of clients, including in the financial industry, where institutions require accurate and up-to-date access to a wide variety of information sources.
- Support for collaborative applications, where again events must be disseminated to multiple users to preserve a common user view - for example, in multiuser games
- Support for a range of fault-tolerance strategies, including the consistent update of replicated data on the implementation of highly available servers.

- support for system monitoring and management, including for example load balancing strategies.

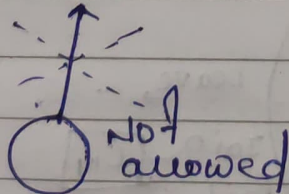
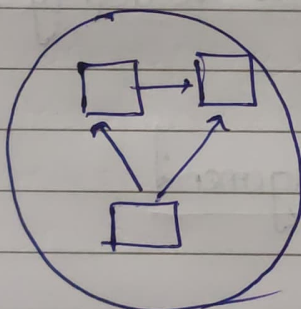
Note

- what are the application of group communication in indirect communication.

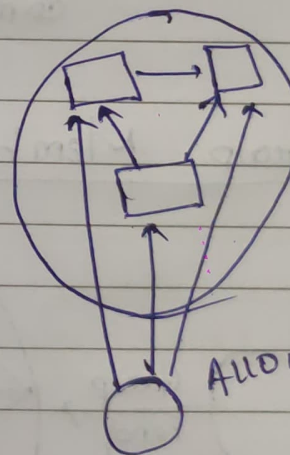


Process Groups and object groups :-

closed group

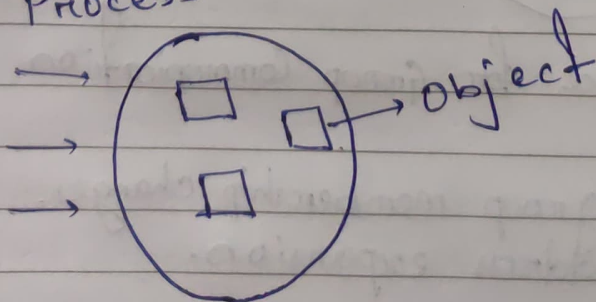


open group



Allowed

Process



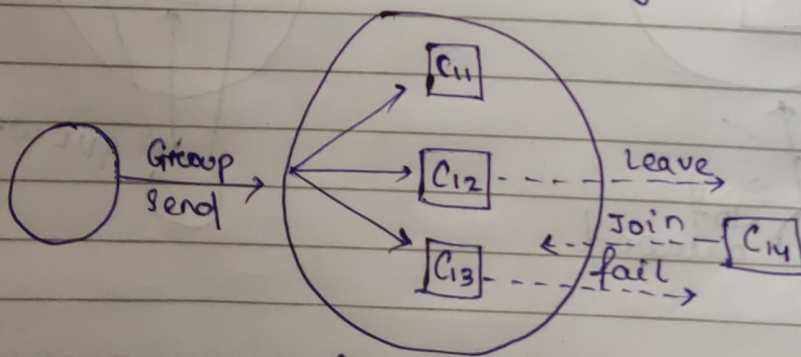
Note

- Q. differentiate the process groups and the object groups.

The programming model

- Part of group communication
- Process and object groups
 - marshalling through communication
 - Not supported by the layers of info
- closed and open groups
- overlapping and non-overlapping groups
- Synchronous and asynchronous systems
 - used to check the validity of group communication

Group Membership Management

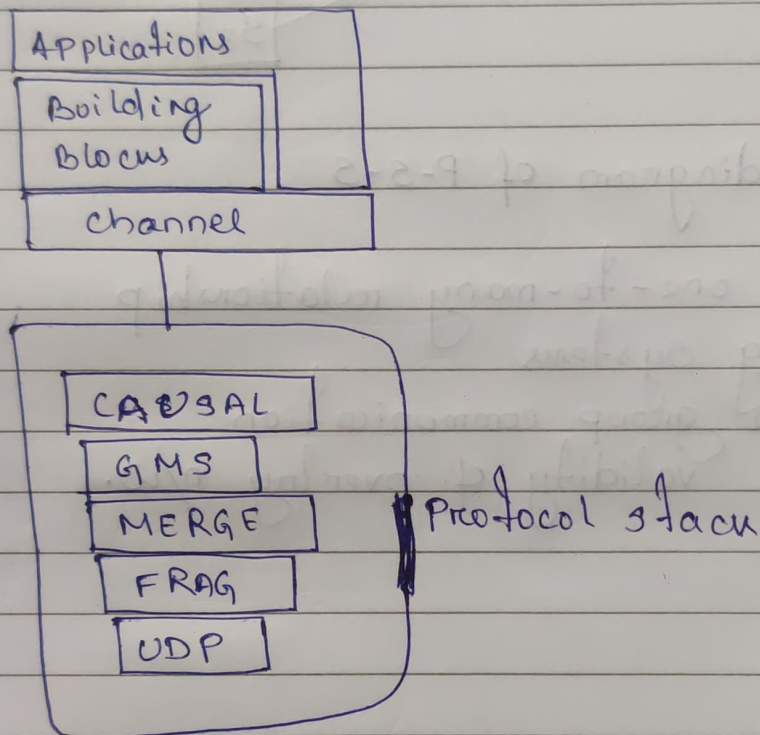


- Provide an interface for Group Communication
- failure detection
- Notify members of group membership changes.
- Performing group address expansion.

Implementation issues recovery techniques :-

- FIFO ordering
- Casual ordering
- Total ordering

J Groups



Channel :- It represents the most primitive interface for application ~~development~~ developers, i.e., joining, leaving, failure of the objects group.

Building Block :- It offers high level abstractions offered by channel.

Protocol stack → It provides the communication protocol

CAUSAL → for ordering

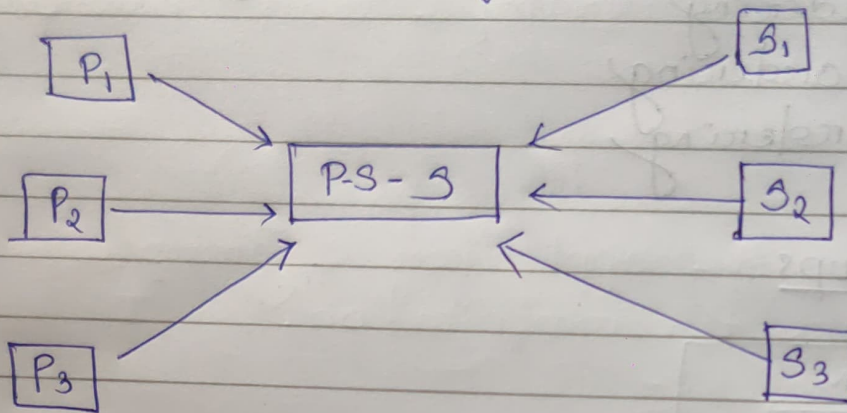
GMS → group membership

MERGE → merging of various communication

FRAG → message fragmentation

UDP → communication protocol

Publish-Subscribe System



Block diagram of P-S-S

- Based on one-to-many relationship
- 4-layering systems
- useful in group communication
- check the validity of overlay n/w.