Introduction

Group communication facilitates the interaction between groups of processes.

Motivation

Important issues

Conventional approaches

Groups of components

Management of groups

Message dissemination

Message delivery

Taxonomy of multicast

Group communication in ISIS

JGroups

Issues

The following section discusses several important basic issues of distributed applications.

Data representation in heterogeneous environments.

Discussion of an execution model for distributed applications.

What is the appropriate error handling?

What are the characteristics of distributed transactions?

What are the basic aspects of group communication (e.g. algorithms used by ISIS)?

How are messages propagated and delivered within a process group in order to maintain a consistent state?

External data representation

Time

Distributed execution model

Failure handling in distributed applications

Distributed transactions

Group communication

Distributed Consensus

Authentication service Kerberos

Motivation

Many application areas such as CSCW profit immensely if primitives for a group communication are supported properly.

Typical application for group communication

Fault tolerance using replicated services, e.g. a fault-tolerant file service.

Object localization in distributed systems; request to a group of potential object servers.

Conferencing systems and groupware.

Functional components (e.g. processes) are composed to a group; a group is considered as a single abstraction.
Important issues of group communication are the following:

**Group membership**: the structural characteristics of the group; composition and management of the group.

**Support of group communication**: the support refers to group member addressing, error handling for members which are unreachable, and the message delivery sequence.

Communication within the group:
- unicasting, broadcasting, multicasting

Multicast messages are a useful tool for constructing distributed systems with the following characteristics:
- fault tolerance based on replicated services.
- locating objects in distributed services.
- multiple update of distributed, replicated data.

**Synchronization**
- the sequence of actions performed by each group member must be consistent.

---

### Group communication

**Introduction**
- Group communication facilitates the interaction between groups of processes.

**Motivation**

**Important issues**

**Conventional approaches**

**Groups of components**

**Management of groups**

**Message dissemination**

**Message delivery**

**Taxonomy of multicast**

**Group communication in ISIB**

**JGroups**

---

### Conventional approaches

**Group addressing**
- Central approach: There is a central group server which knows the current state of the group composition.
- Decentralized approach: Each group member is aware of the group structure and its members.

**Communication services**
- This issue refers to the technology used for the communication between group members.
  - Datagram (for example UDP).
  - Reliable data stream (for example TCP).

In order to get a consistent global group behavior, even in case of errors, a special group communication support is needed, for example ISIS (and the succeeding project Horus) by Cornell University.

---

### Message dissemination

For message dissemination to the group members the following mechanisms are possible options:

- **Unicast**: send and receive messages addressed to individual group members.
- **Group multicast**: send and receive messages addressed to the group as a whole.
- **Inter-group multicast**: send and receive messages addressed to several groups.
- **Broadcast**: send and receive messages addressed to all components (requires filtering).

**Hybrid approach for wide-area networks**
Introduction
Group communication facilitates the interaction between groups of processes.

Motivation
Important issues
Conventional approaches

Groups of components
Management of groups
Message dissemination
Message delivery
Taxonomy of multicast
Group communication in ISIS
JGroups

Atomicity
Atomicity specifies who receives a message.

- in the absence of errors, we have the “exactly-once” semantics, i.e. messages to the group are delivered exactly once to all group members.
- “all-or-nothing” semantics for messages to the group (“atomic broadcast”), i.e. a message is either delivered to all group members or to none.

Atomicity facilitates distributed application programming.

Sequence of message delivery
It is desired to deliver all messages sent to the group G to all group members of G in the same sequence, because otherwise we might get non-deterministic system behavior.

Example for group reconfiguration

Message delivery is an important issue of group communication; two aspects are relevant:

a) who gets the message, and
b) when is the message delivered.

Atomicity

Sequence of message delivery
Ordering for message delivery

m4 is sent by G1 before the group composition is modified. However, in order to guarantee atomicity, m4 should not be delivered to S1 and S2 (since, due to the crash, it is no longer possible to deliver m4 to S3).