GROUP COMMUNICATION SYSTEM SPECIFICATION AND DESIGN FOR NON-REPLICATED SERVICE

Ruiyong Jia

School of Computer Science
Northwestern Polytechnical University ,China

Outline

- Background & Motivation
- GCS Specification for Non-Replicated Service
- Design and Implementation
- Preliminary Performance
- Related Work
- Conclusion

Background & Motivation-1

- Traditional Storage Model

- Direct Attached Storage

- Replicated Service

- Reliability obtained by replicating critical information among multiple clustered servers.

- Examples: replicated database, highly available web service, video-on-demand service, etc.

Background & Motivation-2

- Group Communication System(GCS)

- A kind of middleware introduced for supporting the replicated service.

- GCS Services:

membership,multicast,virtual synchrony.

- The advantages of designing replicated service on top of a GCS:

modularity, simplicity and scalability.

Background & Motivation-3

- Non-Replicated Service

- A kind of server cluster with shared storage

- Example: metadata server cluster in distributed storage system for storage area network.

- Reliability obtained by a direct take-over manner.

- Require lightweight GCS specification：
  membership only.

GCS Specification for Non-Replicated Service-1

- System model：a physical local area network,
  the asynchronous system model
- Lazy failure detection：a processor（Incarnation of GCS daemon）only checks the availability status of other processors on explicit requests.

GCS Specification for Non-Replicated Service-2

- Processor Group Membership Protocol

(Property 1) Self-Inclusion

If a processor p joins group g, then p is a member of g.

(Property 2) Agreement on Group Membership

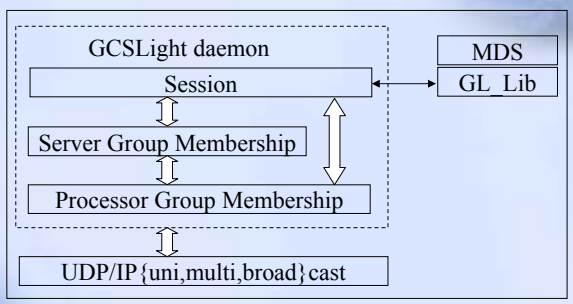
If processor p and q join the same group g, both p and p see the same members in the g.

(Property 3) Agreement on Group History

All processors have the same history.

Design and Implementation-1

1. Architecture



Design and Implementation-2

2.Interfaces

```
GL_connect (const char* GCSLight_name, mailbox * mbox);
GL_disconnect (mailbox * mbox);
GL_join (mailbox * mbox, char * groupname);
GL_leave (mailbox * mbox, char * groupname);
GL_viewcheck (mailbox * mbox, char * error_report);
GL_viewrecv (mailbox * mbox, int max_mess_len, char mess*);
GL_error (int error);
```

Design and Implementation-3

- Algorithm for processor group membership

  The processor group membership algorithm is based on jahanian's work (Jahanian F. et al. 1993). The updating of new membership is done by a 2-phase protocol. What different from jahanian's work is that we adopt a lazy failure detection mechanism in the protocol.
- Algorithm for Server Group Membership

Preliminary Performance

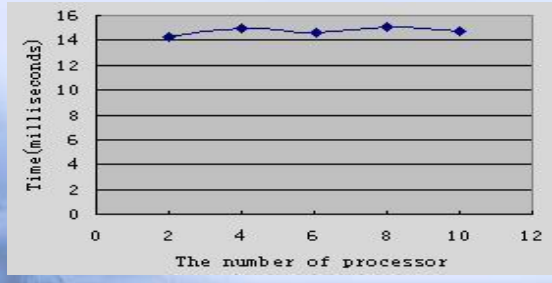
- Environment

  10 Pentium II 350Mhz workstations interconnected by a 10 Megabit/sec Ethernet local area network,Windows 2000 Server as operating system.

- Benchmark

  Join processing time: the time between the start of a processor p and the moment a new group that includes p is formed.

- Result



Related Work

- Most of the existing GCSes were designed for supporting replicated services. For example: ISIS、Phoenix、Transis, Spread, etc.
- They are very complicated and hard to understand.
- Both multicast services and virtual synchrony in these systems are redundant for non-replicated service.
- They use positive failure detection protocol.

Conclusion

- Define the specification of GCS for non-replicated service
- Design a novel group communication system GCSLight according to the specification.
- GCSLight is novel in the following points:

  - provides only the necessary GCS functions for non-replicated service. So it is very lightweight.

  - by adopting a lazy failure detection-based processor group membership protocol, GCSLight itself does not incur any communication overhead in failure-free runs.