TK1 2nd Exercise

Kecen Li, Shule Liu, Zhen Chen, Letian Feng, Xin Zhang

16. November 2016

1 RPC Failure Semantics

The implication from At-Most-Once semantics is that the server will execute the requested procedure only once with its duplicates recognition mechanism when request or reply omission happens, and under circumstance where a crash of the server is encountered, server will not attempt to retry executing the requested procedure and no result will be returned.

As for the server side, requests should be recognised with unique id and be kept track of until receiving an acknowledgement to avoid unnecessary additional executions of procedures when request is omitted. In case of reply omissions, server should also cache execution result for potential request resent from clients.

As for the client side, unique id and timeout should be set for requests so requests could be resent when request or reply omissions happen and recognisable by the server. Client must also send acknowledgements for each reply it receives from the server such that memory occupied by cached result on the server can be freed.

2 Marshalling

a) Marshalling is the process of serialising data structures to be of a unified representation for communication between processes or threads which might be running on different operating system, using different hardware structures and implemented with different programming languages. With marshalling, the data structures which might be represented differently among various systems will now be passed and received as a unified format, thus making message passing feasible between different systems which promote heterogeneity within distributed system.

b) CORBA uses the interface definition language (IDL) to define how data will be presented. When two processes communicating using CORBA, it is assumed that both sides are aware of the underlying structure of the data passed, thus no need to explicitely mention their types.

Pro: Different systems might have different representation of primitive types thus hard to collaborate. With CORBA CDR where there is no explicit typing, they are able to communicate under an agreed data format, thus achieving heterogeneity of the distributed system. In addition, absence of typing information in the messages will lead to reduction in bandwidth used.

Con: Prior knowledge and agreement about the format of the data passed are required for both sending and receiving end, thus incur an overhead during the communication process.

3 Request-Reply Protocols

RR: request-reply, client send request to server and server reply.
RRA: request-reply-acknowledgement, client send request to server,server reply the request,then client acknowledge the reception of reply.

Advantage: with acknowledgements, server is able to cache results of procedures' call and retransmit the result if duplicate request are sent from clients. This is a boost for the server performance. And upon receiving acknowledgement, server is able to free the space occupied by the cached result. On the other side, with RR, server will not be informed whether the reply are received by the client, thus caching becoming impossible as it will consume server memory and making it impossible to scale.

Disadvantage: Sending acknowledgement incurs an overhead for server-client communication thus performing poorer when reply omission is rarely the case.

4 Problems of Concurrency

Distributed system is by nature parallel. Since parallel is an approach to realise concurrency, many concurrent programming concepts can be applied here thus many of the concurrency problems are relevant in distributed system as well.