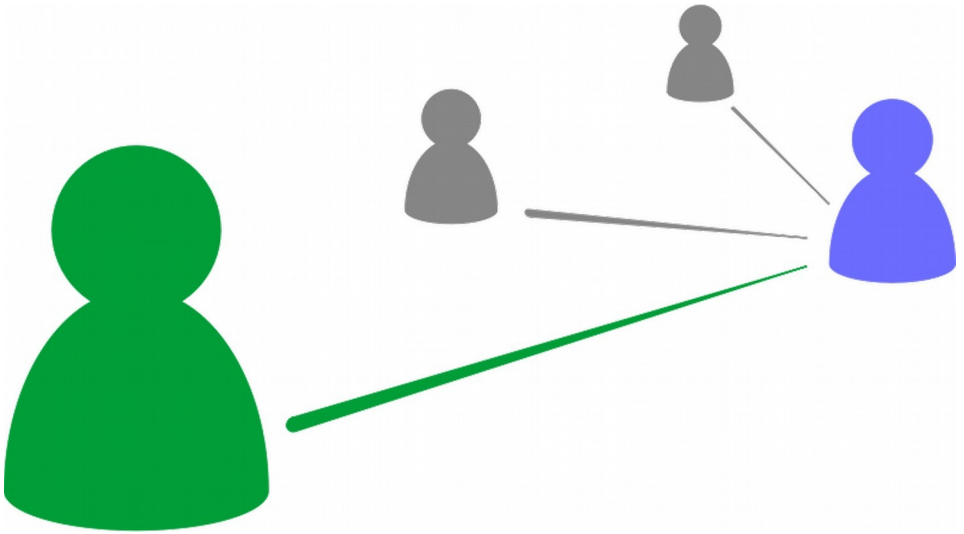
CS 470
Spring 2024

Mike Lam, Professor



Networks (Preview for P3)

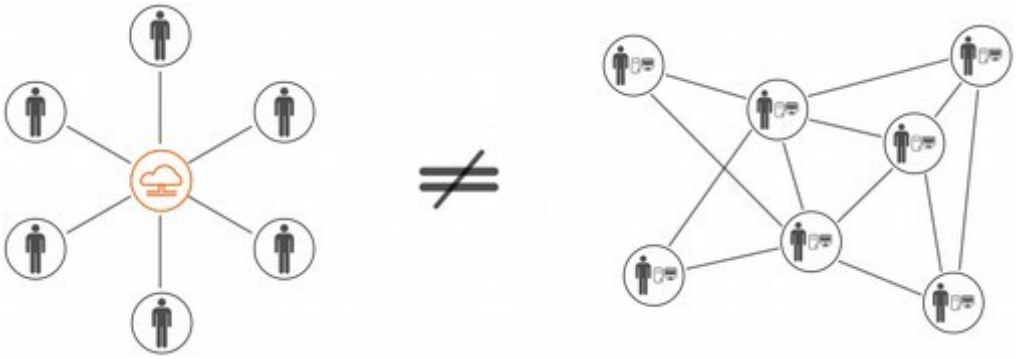
Content taken from IPP 2.3.3 and the following:

"Distributed Systems: Principles and Paradigms" by Andrew S. Tanenbaum and Maarten Van Steen (Chapter 4)

Various online sources (including wikipedia.org and openclipart.org)

Overview

- Topologies – how a network is arranged (hardware)
- Routing – how traffic navigates a network (hardware and software)
- Protocols – how machines communicate (software, low-level)
- IPC paradigms – how processes communicate (software, high-level)



Part 4

- IPC paradigms – how processes communicate (software, high-level)

IPC paradigms

- Inter-process communication (IPC)
  - Message-passing (explicit)
    - Symmetric (SPMD) vs. asymmetric (differentiated hosts)
    - Sockets and MPI
  - Remote procedure calls (implicit)
    - Synchronous vs. asynchronous

Remote Procedure Call (RPC)

- Key idea: transparency
  - It should look like the procedure call is happening locally
  - Similar in spirit to PGAS remote memory accesses
  - Implement server / client stubs to handle the call
- Parameter marshalling
  - Preparing parameters for transmission over a network

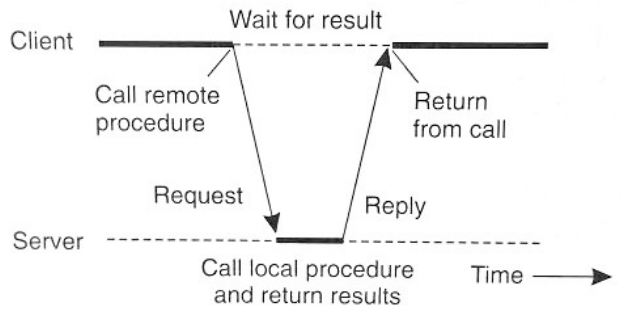


Figure 4-6. Principle of RPC between a client and server program.

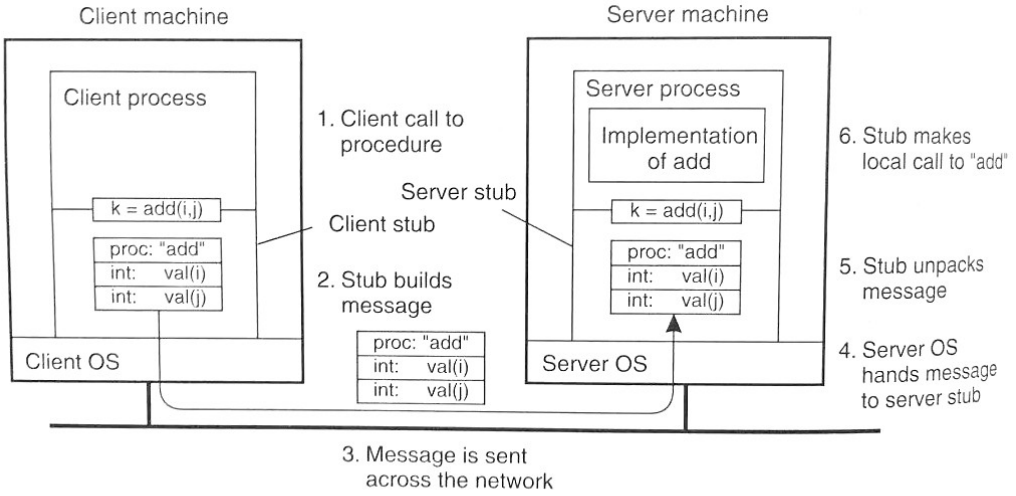


Figure 4-7. The steps involved in a doing a remote computation through RPC.

Asynchronous RPC

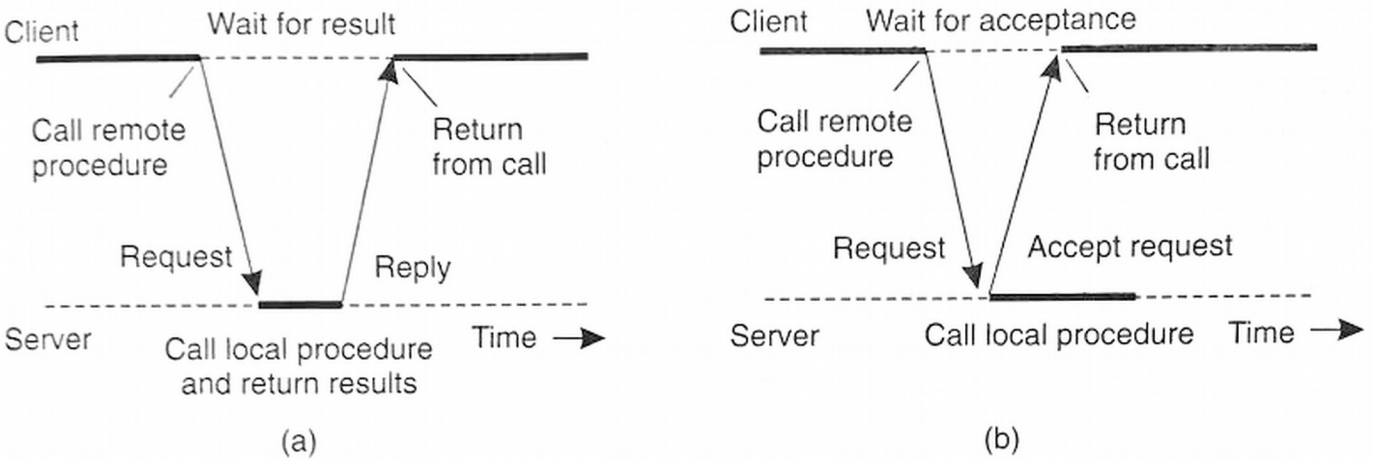


Figure 4-10. (a) The interaction between client and server in a traditional RPC. (b) The interaction using asynchronous RPC.

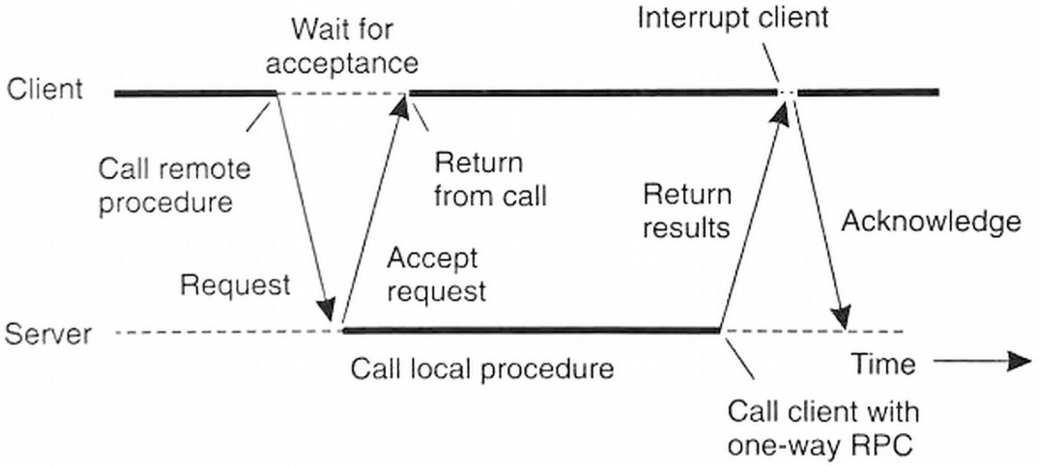


Figure 4-11. A client and server interacting through two asynchronous RPCs.

P3 - DHT

- Distributed hash table
  - n MPI ranks / processes
  - 2 threads per rank
    - Server
    - Client
  - Keys assigned to ranks via provided hash function
  - RPC wrappers for local table operations
    - Rough structure suggested
    - You design the exact protocol

Input file:

```
proc 0

put xyz 5
put bar 99
put z 7
sync
get xyz
get blah
get qw
size

proc 1

put baz 1
put stuff 2
sync
```