5 - Client-Server Anatomy

Tutor version

Dr Leonardo Mostarda,
School of Science and Technology,
Camerino University, London

Version 1.0, 10 Mar 2013

Question 1

Proxies can support replication transparency by invoking each replica. Can (the server side of) an application be subject to a replicated calls?

Solution 1

Yes: consider a replicated object A invoking another (nonreplicated) object B. If A consists of k replicas, an invocation of B will be done by each replica. However, B should normally be invoked only once. Special measures are needed to handle such replicated invocations.

Question 2

Constructing a concurrent server by spawning a process has some advantages and disadvantages compared to multithreaded servers. Mention a few.

Solution 2

An important advantage is that separate processes are protected against each other, which may prove to be necessary as in the case of a superserver handling completely independent services. On the other hand, process spawning is a relatively costly operation that can be saved when using multithreaded servers. Also, if processes do need to communicate, then using threads is much cheaper as in many cases we can avoid having the kernel implement the communication.

Question 3

Sketch the design of a multithreaded server that supports multiple protocols using sockets as its transport-level interface to the underlying operating system.

Solution 3

A relatively simple design is to have a single thread T waiting for incoming transport messages (TPDUs). If we assume the header of each TPDU contains a number identifying the higher-level protocol, the tread can take the payload and pass it to the module for that protocol. Each such module has a separate thread waiting for this payload, which it treats as an incoming request. After handling the request, a response message is passed to T, which, in turn, wraps it in a transport-level message and sends it to tthe proper destination.

Question 4

Is a server that maintains a TCP/IP connection to a client stateful or stateless?

Solution 4

Assuming the server maintains no other information on that client, one could justi?ably argue that the server is stateless. The issue is that not the server, but the transport layer at the server maintains state on the client. What the local operating systems keep track of is, in principle, of no concern to the server.

Question 5

Imagine a Web server that maintains a table in which client IP addresses are mapped to the most recently accessed Web pages. When a client connects to the server, the server looks up the client in its table, and if found, returns the registered page. Is this server stateful or stateless?

Solution 5

It can be strongly argued that this is a stateless server. The important issue with stateless designs is not if any information is maintained by the server on its clients, but instead whether that information is needed for correctness. In this example, if the table is lost for what ever reason, the client and server can still properly interact as if nothing happened. In a stateful design, such an interaction

Question 6

In order to implement a distributed server we can make use of mobility support for IP version 6 (MIPv6). Discuss the use of route optimisation.

Solution 6

Home agent and Care of Address are a centralised point of failure. Route optimisation can solve the problem.