NETWORK INTERFACES OF CLIENT-SERVER LEVEL

Yu. Morozov, I. Pasternak

Lviv Polytechnic National University, Computer Engeneering Department

It is clear that, in general, in order the application program running on the workstation can request a service from a server, at least is required interface software level that supports such kind of interaction (it would be at least unnatural to require that the application program directly use the primitive transport level of local network). And it causes the basic principles of the system architecture "client-server".

The system is divided into two parts that can be executed in different nodes of the network - client and server parts. An application program or a final user interact with the client part of the system, which in the simplest case provides just over networks interface. The client part of the system when it is needed makes request through the network to the server part. It is noticed, that in the developed systems a network request to the server part may not be necessary if the system can predict the user needs and the client part contains data that can meet user next request.

The interface of server part is defined and fixed. So it is possible to create new client parts of the existing system (an example of interoperability at the system level).

The main problem of systems based on the architecture "client-server", is that in accordance with the concept of open systems is required theirs mobility in as wide as possible class of hardware and software solutions of open systems. Even in case of limit to UNIX-oriented local networks, in different networks are used different apparatus and communication protocols. Attempts to create systems that support all possible protocols lead to theirs overload and damage to functionality. Even more challenging aspect of this problem is associated with the ability to use different data representations on different nodes in a heterogeneous local network. Different computers may have different addressing, numbers presentation, character encoding, etc. It is especially significant for servers of high level: telecommunication, computing, databases.

A common solution of the mobility problem of systems based on the architecture "client-server" is the reliance on software packages that realize the protocols of remote procedure call (RPC - Remote Procedure Call). When using such means of request to the service in the remote node looks like a normal procedure call. RPC means, which, naturally, contain all the information about the specificity of the aparatus of local networks and network protocols, change the call into a succession of network interactions. Thus, the specificity of network environment and protocols is hidden from the applied programmer.

When calling the remote procedure call of RPC program is performed the conversion of client data formats in intermediate machine-independent formats and then convertion in the data formats of the server. When transmitting the appropriate parameters similar transformations are produced. If the system is implemented on the basis of standard RPC package, it can be easily transfered to any open environment.

The network interface it is two or more PCs connected with the aim quickly to exchange data and share use of resources. To implement the network interface are required two kinds of components: hardware and software. The hardware provides the physical connection of computers. It is divided into local and global networks. For the organization of a network interface it is needed to have network software, transmission physical medium and switching devices. The main results of the load testing is the calculation of the indices of network interface productivity such as the response time of the server to the client request and the overload of system resources of the server, depending on the number of users. In accordance with results can be carried out modification of the hardware and software configuration of the network interface.

The implementation of client-server interaction based on the model of request/response in the form of formulas is suggested. As well as testing of the network interface on the load of client requests.

Keywords - network, software, network testing.