

Distributed Systems Assessed Coursework

Lecturer: M. Sloman

Hand-in Date: Tuesday 16 March 2010

Question

The purpose of the exercise is to compare Java RMI and UDP for their relative reliability and ease of use.

For each case you will have to write a simple client that passes a specified number of messages to a server. Each message should contain a message sequence number and the total number of messages to be sent. The server keeps track of the messages received and when there are no more messages, outputs a summary of the number of messages received and also which ones were lost.

When the clients and servers for each communication mechanism are working, run some experiments on 2 computers *in different parts of the lab (ie not physically near each other)*, sending increasing numbers of messages (e.g. from 20-100 with increments of 20, then, 200, 300 and 400) and identify the situations in which messages are lost.

Do not send more than about 1000 messages as otherwise you will overload the lab network.

You should hand in the following as hardcopy:

1. A short summary (no more than 1 page) describing your findings from running the programs. You should address the following points:
 - a. For each mechanism, what are possible causes, if any, of messages being lost?
 - a. Are there any patterns in the way messages are lost?
 - b. What is the relative reliability of the different communication mechanisms?
 - c. Which was easier to program and why.
 - d. Proof that the both RMI and UDP programs actually ran e.g. console logs or screen dump plus an indication of which message numbers, if any, were lost.
2. A well formatted listing of the completed code for the 4 classes (2 client server pairs) which is easy to read. Start each of the 4 classes on a new page and avoid very long lines (i.e. <70 characters). Your solutions should deal with exceptions appropriately.

Please provide program listings in the following order

RMI Client

RMI Server

UDP Client

UDP Server

/ PTO

Notes:

Download outline software and scripts from
<http://www.doc.ic.ac.uk/~mss/DScw/install.html>

Some source code outlines and supporting files are provided. You are not obliged to use these; however they should simplify the process of achieving working solutions. A few notes on the layout and support files follow:

After extracting the files from the archive provided, you should run `install.csh` (or `install.bat` on Windows) to obtain the appropriate build and execution scripts which are described below.

The class `MessageInfo` (in the *common* folder) provides a container for the data to be sent and also has a constructor that extracts the data from a string representation. Outline code for each of the client/server pairs can be found in the *rmi*, *udp* and *tcp* folders.

The file "policy" is a simple configuration file required for the RMI code. More constrained policies are possible, but this should provide the lowest barrier to testing.

The Makefile allows Linux users to use `make` to compile the various parts of the exercise. It can also be used to help configure your preferred development environment with the correct commands, flags and parameters. Windows users can use the `build.bat` script to compile the exercise in the same way. The other shell scripts (`rmiclient` (.bat or .csh) etc) allow users to execute the various parts of the exercise.

Tutorials

UDP: <http://java.sun.com/docs/books/tutorial/networking/sockets/>

RMI: <http://java.sun.com/docs/books/tutorial/rmi/index.html>

<http://developer.java.sun.com/developer/onlineTraining/rmi/RMI.html>