

ASSIGNMENT 4: Internet Game

Out: 3/7/02; Due: 3/21/02

Programming Parallel and Distributed Systems

Computer Science 178, Spring 2002

Steven P. Reiss

OBJECTIVE

The purpose of this assignment is to let you develop and play with an internet application. This means developing appropriate interactive web pages, code to cooperate in the web server, and an appropriate back end.

THE PROBLEM

Rather than develop a new internet application from scratch, we are going to take the game that you developed for the last assignment and adapt it to the internet. This should make the assignment somewhat simpler. It will also stress the differences between internet-based applications and client-server applications since you will have to adapt your code to these differences.

SPECIFICATIONS

Your job then is to adapt your client-server application to the Internet. To do this you will first have to decide how the game can be played through the browser. You can do this either through a sequence of pages or through modifications to the existing page. In theory, this could be done by simply converting your client application into an applet, but I would consider this cheating and will require you to use javascript and html for your web pages. To understand what is possible and desirable here, I strongly recommend that you prototype these pages before starting to think about the middleware or modifications that will have to be made to your server. You should think about user interface issues when designing the web pages — try to make the user interaction as smooth and intuitive as possible.

Once you have designed a web-based front end for your application, you can go ahead and implement the middle and back end. We will be providing a dedicated web server that will offer you a writable cgi-bin as well as java server pages and servlets. Details on how to access this, where the web directories reside, and what needs to be done to add or update servlets or jsp pages will be provided in the newsgroup. You can use any of these technologies for the middleware portion of your system. You should decide what if any functionality should go in the middleware. It is possible that you will want to migrate some of your original client functionality to here. It is also possible that you will want to simplify the actual server by splitting it so that a portion resides in the

middleware. You can continue to use Java RMI to communicate from the middleware to the server, but you may also choose another appropriate technology if you wish.

Note that you should adapt your server so that it does not assume that clients are connected all the time. You will be able to detect most instances of clients going away (there are javascript hooks for leaving a web page), but not all. Note also that you will have to provide front end web pages both for connecting to an appropriate game and for playing the game. Unlike the client-server assignment, we will assume that there is only a single server to further simplify matters.

The particular functionality we will be looking for includes:

- A web page that shows the user all the active games and lets the user choose among them. If possible, this should update dynamically as new games are started. If you can't figure out how to do this automatically, at least provide an update button on the page to let the user see the current set of games.
- One or more web pages for playing the game. These should use HTML and Javascript to implement a high-quality, appropriate client interface to your game.
- Middleware, in the form of JSP, servlets, or cgi-bin scripts, that connects your set of front end pages to your server.
- Your game server from the previous assignment appropriately modified to handle internet clients.
- Robustness of the middleware and server in the face of clients coming and going. Here you might want to assign session identifiers to the client pages so that they can communicate appropriately. You will then need either the middleware or server manage these identifiers.

You might also want to include lots of error checking throughout to ensure that the messages you are sending back and forth are appropriate. For debugging purposes, you might want either the server or the middleware to create logs of all the events they receive.

TESTING

Because this assignment will require multiple browsers accessing the server at once, it will be more difficult to test. You will probably want to open multiple browser windows on one or more machines and try playing your game this way. Ideally you should make sure that your web pages work on both IE and Netscape.

MECHANICS

You should hand in full source code and runnable binaries of your server. The middleware and web pages should be installed on the web server that is set up for the class; you should hand in the sources for these as well.