Networking: Game2D Code Study

Tom Kelliher, CS 245

Oct. 7, 2002

1 Administrivia

Announcements

Deliverables on Wednesday. Exam on Friday.

Assignment

None.

From Last Time

Threads and Synchronization.

Outline

1. Hints for developing network code.

2. Game2D code study.

Coming Up

Networking Lab.

2   Hints for Developing Network Code

1. Currently, network applets must be run from inside the firewall.

   Requesting firewall access for a block of ports on black.

2. Sun's AppletViewer's implementation of the Socket library leaves much to be desired.
   And you have to use the policytool to seriously weaken Java's security model.

   What you'll need to do: Serve your applet from (soon to be installed) Apache on black.
   Java's security model permits an applet to connect back to the server which served it.

   It may be easier to do you client development on black.

3. If you re-compile your applet for serving from a web server, you must exit and re-start
   your browser so that it loads fresh versions of the class files.

3   Game2D Code Study

Why I'm not putting the source code on the class web page.

Work in your teams to answer the following:

1. `Game2Dclient.java` questions:

   (a) Each client runs two threads. What are they?

   (b) Once the game has begun, what action or actions cause the applet class (line 8)
       to send a message to the server? What is the format of those messages?

   (c) Where in the applet class code is the connection to the server made?

   (d) The listen class (line 130) only receives messages from the server. What kinds
       of messages does it receive? Does it act on those messages directly or indirectly
       (through a synchronized queue or some other such mechanism.).

2. `G2DLaunch.java` questions:

(a) Why is the main class so short (lines 276–282)?

(b) Looking at line 20, explain what the Timer does and how it does it. It may be helpful to have a look at lines 202–204.

(c) Explain how the G2DServer class (line 5) design permits multiple games to occur simultaneously.

(d) Three server threads are required per game. What are they?

(e) What are the main responsibilities of the Game2D class (line 56)?

(f) In line 43, why is the server connecting to itself? Isn't that illegal in Maryland?

(g) Do the ThreadedEchoHandlers (line 137) communicate directly with the two clients or indirectly (using another thread or each other as agents)?

(h) Why is the sky blue?

3. General question: Where are potential synchronization problems?