Sockets and Networking

12-2-2004

Opening Discussion

- What did we talk about last class? Do you have any code to show?
- Do you have any questions about the assignment?
- What do you know about networking of computers?

Code for Streams

- Before we start talking about networking in Java, let's finish the code we started last time dealing with streams in Java. It will be significant for today.
- We are writing a DrawText class that includes Open and Save options for text files.
- We also want to use Serialization and writing of objects to file so we can save whole drawings.

Motivation

- This is an easy one to motivate. We want to be able to have computers "talk" to one another. You do this all the time these days and don't even realize it. Ten years ago it happened a lot less and wasn't as transparent to the user.
- It still isn't completely transparent to the programmer and you have to do special things when you want to talk to another computer.

Networking

- The networking functionality in Java is mostly contained in the java.net package and that is what we will focus on today. Next class we will look at another way of doing networking that can be easier for large applications.
- The java.net package has a nice collection of classes to help you do communication between computers.

Sockets

- The way we do communication between machines is through things called sockets. We try to establish a socket between two computers and send information over it.
- Sockets come in different flavors though and there are several types in the java.net package.
- Sockets generally support the client/server model of communication.

TCP Sockets

- Most network communication is done through the Transfer Control Protocol (TCP). The ServerSocket and Socket classes in Java use this. It guarantees that data gets across or tells you when it doesn't.
- To do the communication, one machine sets up a ServerSocket on a particular port and waits for a connection using accept(). Another program can connect to that port with a Socket. Sockets should be closed when you are done using them as they take up system resources.

Talking Through Sockets

- The ServerSocket accept method returns a Socket itself so each machine has a socket object to play with.
- The Socket class has getInputStream() and getOutputStream() methods. Once we have these streams we can use them just like we would a stream to a file. In this case though we are sending information to a different computer.

URLs

- The java.net package also provides a handy class called URL. This class allows you to easily interact with resources on the world wide web.
- Here again, there is a method called openStream that gives you an input stream that can be used to read the contents of whatever is at the URL. You can also get a URLConnection.

Datagrams

- In addition to TCP, there is also a User Datagram Protocol (UDP). This protocol is unreliable in the sense that it doesn't let you know if a message doesn't make it through. However, it is much faster than TCP because of this. For that reason, you use it if you need speed and can recover from missing a message or two.
- The DatagramSocket and DatagramPacket classes are used for this.

Code

- Let’s try to create some code that will allow two users to communicate using our draw program.

Minute Essay

- What type of applications would you want to use UDP for? What types would you distinctly not want to use UDP for and want to stick with TCP instead?
- Remember that assignment #7 is due today.