CS 550:02: Coding Project #1

Due on Sunday, January 29, 201

Lan 11:25am

Neil Getty & Christopher Hannon

Test Case 1

To start the environment use the command: sudo mn topo=single,4
After Mininet loads, create a terminal for each host: xterm h1 h2 h3 h4

Listing 1: Sample Input and Output

```
user@user-VirtualBox:~/iit-cs550-pa1/src$ sudo mn --topo=single,4

*** Creating network
*** Adding controller
*** Adding hosts:
h1 h2 h3 h4
*** Adding switches:
s1
*** Adding links:
(h1, s1) (h2, s1) (h3, s1) (h4, s1)
*** Configuring hosts
h1 h2 h3 h4
*** Starting controller
c0
*** Starting 1 switches
s1 ...
*** Starting CLI:
mininet> xterm h1 h2 h3 h4
```

Host 1 has IP address 10.0.0.1, Host 2 has IP Address 10.0.0.2 and soforth
In terminal 1, start the index server:

Listing 2: Start the Index Server

```
rmiregistry &
java main.java.host.ServerImpl
```

In terminals 2-4, start the Peers

Listing 3: Start the Peers

```
rmiregistry &
java main.java.peer.WatchDir directory 10.0.0.HOSTID &
java main.java.peer.ClientDriver 10.0.0.HOSTID directory
```

The correct output of the server looks like:

Listing 4: Output of the server

```
INFO: initiating server deamon
INFO: server initiated.

INFO: Press return Key to Exit.


INFO: registering file: songFile.mp3 to peer 10.0.0.3
```

Test Case 1 continued on next page...

```
INFO: registering file: texFile.tex to peer 10.0.0.3
INFO: registering file: pictureFile.jpg to peer 10.0.0.3
INFO: registering file: orgFile.org to peer 10.0.0.2
INFO: registering file: movieFile.avi to peer 10.0.0.2
INFO: registering file: docFile.doc to peer 10.0.0.2
INFO: registering file: file1.txt to peer 10.0.0.4
INFO: registering file: file7.txt to peer 10.0.0.4
INFO: registering file: file5.txt to peer 10.0.0.4
INFO: registering file: file3.txt to peer 10.0.0.4
INFO: registering file: file6.txt to peer 10.0.0.4
INFO: registering file: file2.txt to peer 10.0.0.4
INFO: registering file: file8.txt to peer 10.0.0.4
INFO: registering file: file9.txt to peer 10.0.0.4
INFO: registering file: file4.txt to peer 10.0.0.4
INFO: registering file: file10.txt to peer 10.0.0.4
INFO: registering file: newFile.txt to peer 10.0.0.4
```

The server registers each file from the peer using registerAll().
The correct output of the Client looks like:

Listing 5: Output of the Client

```
INFO: Initializing Peer...
INFO: Client Process initialized...
INFO: Registering Files in: ./test3/
INFO: registering file: file1.txt
INFO: registering file: file7.txt
INFO: registering file: file5.txt
INFO: registering file: file3.txt
INFO: registering file: file6.txt
INFO: registering file: file2.txt
INFO: registering file: file8.txt
INFO: registering file: file9.txt
INFO: registering file: file4.txt
INFO: registering file: file10.txt
INFO: Files Sucessfully Registered...

Input 'exit' to close the application at anytime

Input name of file you want to retrieve:
```

The Client registers each file upon starting. When we create a new file the files are registerd automatically using the watch directory java process. Creating a new file with 'touch test3/newFile.txt' creates the last line in the server log.

When host 3 requests file10.txt (a 1MB file), the server allows the peer to download the file and then it is registered to server.

Listing 6: Output of the server after downloading a file

```
INFO: initiating server deamon
INFO: server initiated.
```

Test Case 1 continued on next page...

```
INFO: Press return Key to Exit.


INFO: registering file: songFile.mp3 to peer 10.0.0.3
INFO: registering file: texFile.tex to peer 10.0.0.3
INFO: registering file: pictureFile.jpg to peer 10.0.0.3
INFO: registering file: orgFile.org to peer 10.0.0.2
INFO: registering file: movieFile.avi to peer 10.0.0.2
INFO: registering file: docFile.doc to peer 10.0.0.2
INFO: registering file: file1.txt to peer 10.0.0.4
INFO: registering file: file7.txt to peer 10.0.0.4
INFO: registering file: file5.txt to peer 10.0.0.4
INFO: registering file: file3.txt to peer 10.0.0.4
INFO: registering file: file6.txt to peer 10.0.0.4
INFO: registering file: file2.txt to peer 10.0.0.4
INFO: registering file: file8.txt to peer 10.0.0.4
INFO: registering file: file9.txt to peer 10.0.0.4
INFO: registering file: file4.txt to peer 10.0.0.4
INFO: registering file: file10.txt to peer 10.0.0.4
INFO: registering file: newFile.txt to peer 10.0.0.4
INFO: registering file: file10.txt to peer 10.0.0.3
```

to prove that the files are tranfered successfully run sdiff test3/file10.txt test2/file10.txt
Only the differences are printed if there was a problem in the transfer.

Test Case 1 continued on next page...