A Simple Peer-to-Peer File Sharing System

User Manual

Allen VanNoppen Jr.
A20415658

Step 1: Download and Prep

1. Move the folder to the desired folder and unzip if necessary.
2. Open up terminal (mac) or command prompt (windows)
3. Type in: “cd” followed by the path to where you placed the program.
    a. Ex “cd Documents/”IIT Graduate Program”/CS550/Program/src1

Step 2: Start Index Server

1. Type in “ls” and you should see the program files listed.
2. For mac type “rmiregistry 1099 &” or windows “start rmiregistry 1099”

```
Allens-MacBook-Pro:src1 allenvannoppenjr$ rmiregistry 1099 &
[1] 29070
Allens-MacBook-Pro:src1 allenvannoppenjr$
```

3. type “rmic IndexServer” (disregard the warning)

```
Allens-MacBook-Pro:src1 allenvannoppenjr$ rmiregistry 1099 &
[1] 29070
Allens-MacBook-Pro:src1 allenvannoppenjr$ rmic IndexServer
Warning: generation and use of skeletons and static stubs for JRMP
is deprecated. Skeletons are unnecessary, and static stubs have
been superseded by dynamically generated stubs. Users are
encouraged to migrate away from using rmic to generate skeletons and static
stubs. See the documentation for java.rmi.server.UnicastRemoteObject.
Allens-MacBook-Pro:src1 allenvannoppenjr$
```

4. type “java StartIndexServer 1099” into the command prompt. You should see the below message. “index server is ready!”

```
Allens-MacBook-Pro:src1 allenvannoppenjr$ rmiregistry 1099 &
[1] 29070
Allens-MacBook-Pro:src1 allenvannoppenjr$ rmic IndexServer
Warning: generation and use of skeletons and static stubs for JRMP
is deprecated. Skeletons are unnecessary, and static stubs have
been superseded by dynamically generated stubs. Users are
encouraged to migrate away from using rmic to generate skeletons and static
stubs. See the documentation for java.rmi.server.UnicastRemoteObject.
Allens-MacBook-Pro:src1 allenvannoppenjr$ java StartIndexServer 1099
Index Server is Ready! Peers may register.
```

Step 3: Connect Peer1 to IndexServer:

1. Open another window in terminal or the command prompt.
2. Again using the “cd” command navigate to the program files and “/Peer1”
3. Type “rmiregistry 5001 &”

Note: peer port number needs to be different from indexserver.

```
Allens-MacBook-Pro:Peer1 allenvannoppenjr$ rmiregistry 5001 &
[1] 29165
Allens-MacBook-Pro:Peer1 allenvannoppenjr$
```

4. type “rmic PeerClient"

```
Allens-MacBook-Pro:Peer1 allenvannoppenjr$ rmiregistry 5001 &
[1] 29165
Allens-MacBook-Pro:Peer1 allenvannoppenjr$ rmic PeerClient
Warning: generation and use of skeletons and static stubs for JRMP
is deprecated. Skeletons are unnecessary, and static stubs have
been superseded by dynamically generated stubs. Users are
encouraged to migrate away from using rmic to generate skeletons and static
stubs. See the documentation for java.rmi.server.UnicastRemoteObject.
Allens-MacBook-Pro:Peer1 allenvannoppenjr$
```

5. type “java StartPeerClient 1099 5001 Peer1

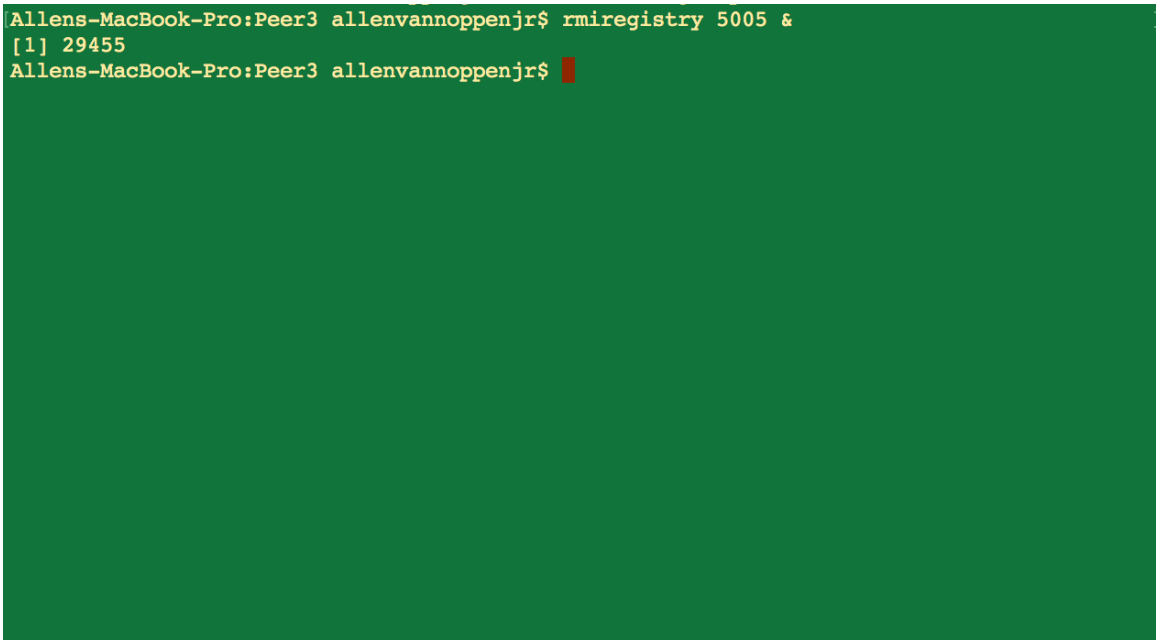
This should connect you to the server and show the files in your directory. The machine now has the Index Server and a Peer running on it. If you so choose you can download the program onto a separate machine and follow the same steps in Step 3 to set up another peer.

```
Allens-MacBook-Pro:Peer1 allenvannoppenjr$ rmiregistry 5001 &
[1] 29165
Allens-MacBook-Pro:Peer1 allenvannoppenjr$ rmic PeerClient
Warning: generation and use of skeletons and static stubs for JRMP
is deprecated. Skeletons are unnecessary, and static stubs have
been superseded by dynamically generated stubs. Users are
encouraged to migrate away from using rmic to generate skeletons and static
stubs. See the documentation for java.rmi.server.UnicastRemoteObject.
Allens-MacBook-Pro:Peer1 allenvannoppenjr$ java StartPeerClient 1099 5001 Peer1
null:
file3.txt
PeerFunction.class
StartIndexServer.class
.DS_Store
file5.txt
PeerClient.class
PeerClient_Stub.class
StartPeerClient$peerfunction.class
PeerFunctionInterface.class
IndexServer.class
IndexWatch.class
PeerClientInterface.class
IndexServerInterface.class
StartPeerClient.class
file8.txt
F250 INSURANCE CARD.pdf
IndexServer_Stub.class = added
Successfully connected!!
File name:
```

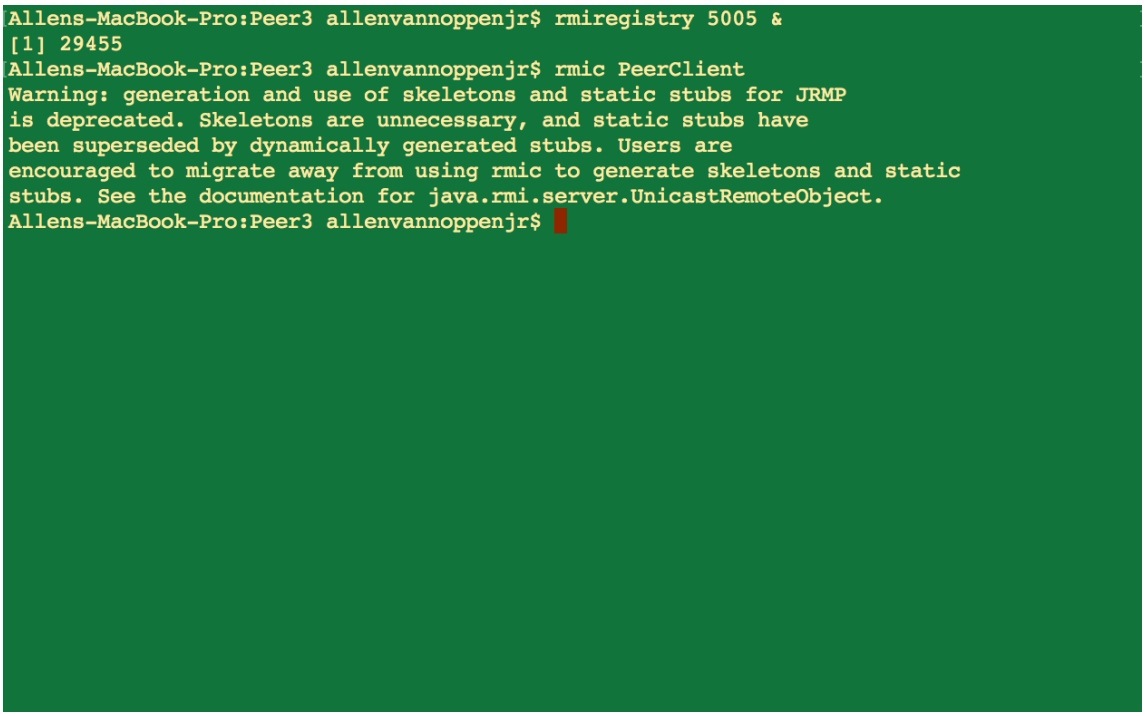
Step 4: Register Peer 2

Not much fun to send files to yourself so navigate to Peer 2 directory and follow the similar steps below.

1. type “rmiregistry 5001 &”



2. type “rmic PeerClient”



3. type “java StartPeerClient 1099 5005 Peer2”

```
Allens-MacBook-Pro:Peer3 allenvannoppenjr$ rmiregistry 5005 &
[1] 29455
Allens-MacBook-Pro:Peer3 allenvannoppenjr$ rmic PeerClient
Warning: generation and use of skeletons and static stubs for JRMP
is deprecated. Skeletons are unnecessary, and static stubs have
been superseded by dynamically generated stubs. Users are
encouraged to migrate away from using rmic to generate skeletons and static
stubs. See the documentation for java.rmi.server.UnicastRemoteObject.
Allens-MacBook-Pro:Peer3 allenvannoppenjr$ java StartPeerClient 1099 5005 Peer3
null:
PeerFunction.class
StartIndexServer.class
.DS_Store
file7.txt
PeerClient.class
PeerClient_Stub.class
StartPeerClient$peerfunction.class
PeerFunctionInterface.class
IndexServer.class
IndexWatch.class
PeerClientInterface.class
IndexServerInterface.class
StartPeerClient.class
file8.txt
file9.txt
IndexServer_Stub.class = added
Successfully connected!!
File name:
```

You can now send files back and forth!