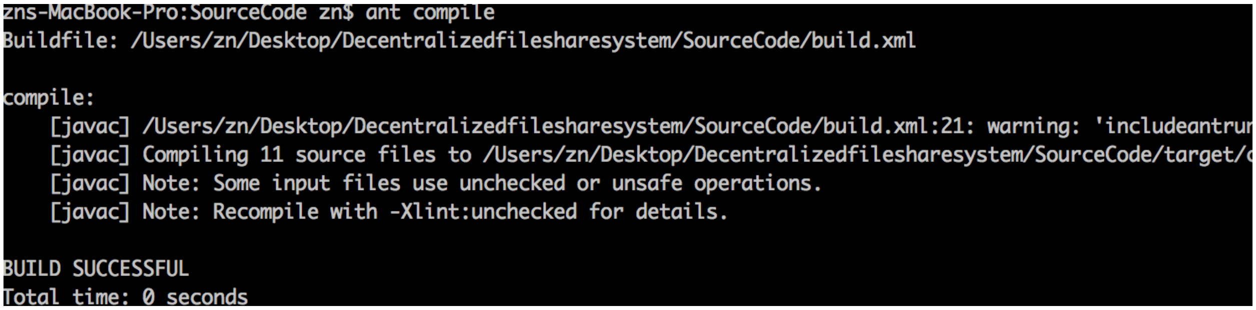
Manual

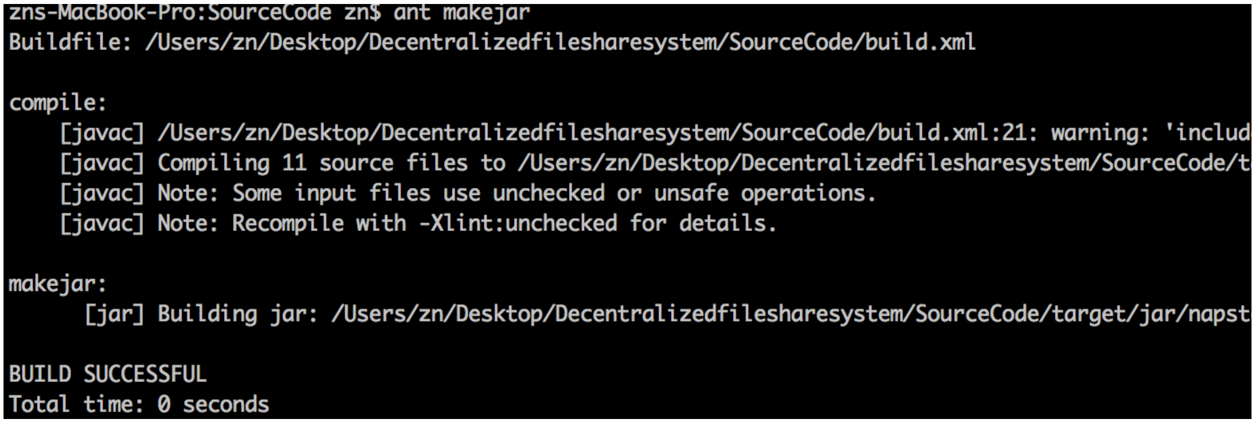
Name: Ning Zhang
CWID: A20336916
Department of Computer Science, Illinois Institute of Technology

Step1.Firstly, you should enter the SourceCode directory. And entering the command 'ant compile' in terminal to use ant script which is called as 'build.xml' in the SourceCode directory to compile the code.



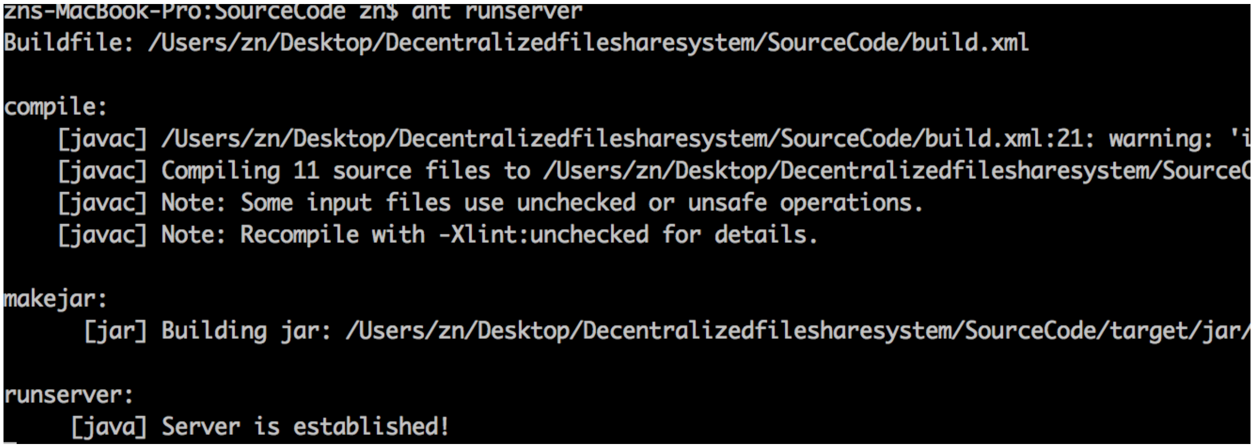
```
zns-MacBook-Pro:SourceCode zn$ ant compile
Buildfile: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/build.xml

compile:
    [javac] /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/build.xml:21: warning: 'includeantrun
    [javac] Compiling 11 source files to /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/c
    [javac] Note: Some input files use unchecked or unsafe operations.
    [javac] Note: Recompile with -Xlint:unchecked for details.

BUILD SUCCESSFUL
Total time: 0 seconds
```

Step 2. Then entering the command 'ant makejar' in terminal to generate the .jar file.



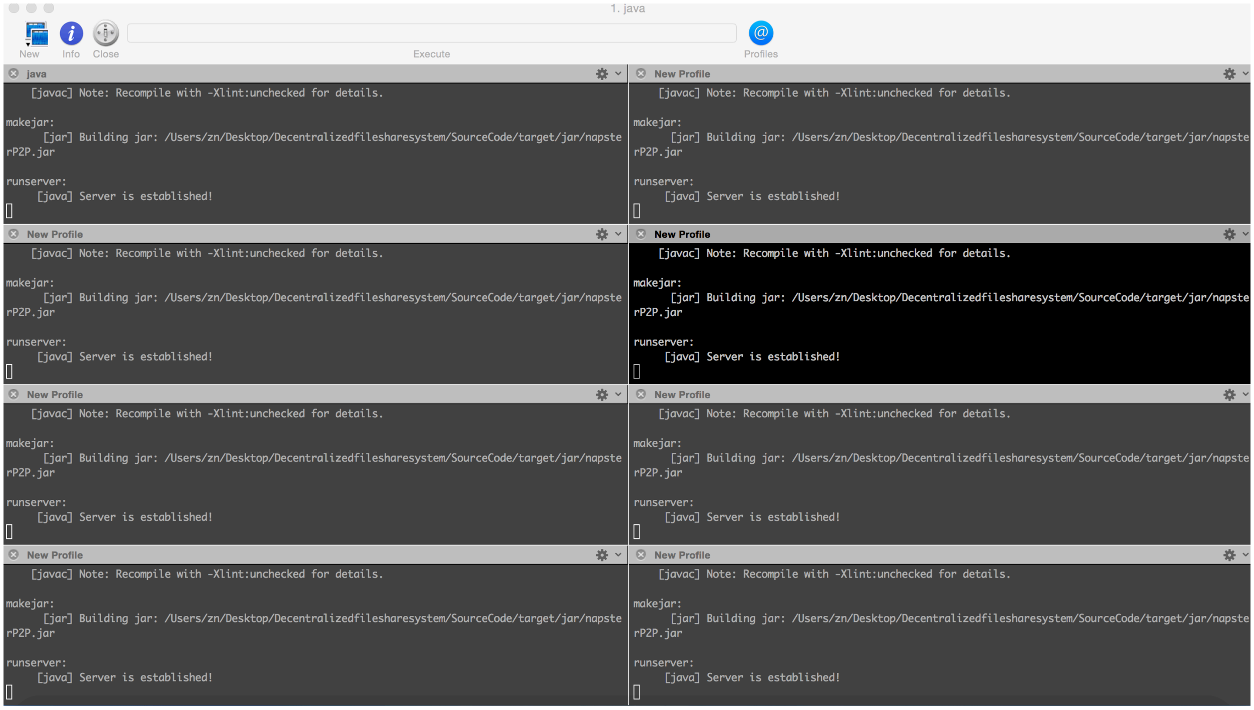
```
zns-MacBook-Pro:SourceCode zn$ ant makejar
Buildfile: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/build.xml

compile:
    [javac] /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/build.xml:21: warning: 'includ
    [javac] Compiling 11 source files to /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/t
    [javac] Note: Some input files use unchecked or unsafe operations.
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napst

BUILD SUCCESSFUL
Total time: 0 seconds
```

Step3. Then entering the command 'ant runserver' in terminal to run the Decentralizedservrer class which is the IndexServer. We can see the outputs in the following screenshot.



```
zns-MacBook-Pro:SourceCode zn$ ant runserver
Buildfile: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/build.xml

compile:
    [javac] /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/build.xml:21: warning: 'i
    [javac] Compiling 11 source files to /Users/zn/Desktop/Decentralizedfilesharesystem/SourceC
    [javac] Note: Some input files use unchecked or unsafe operations.
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/

runserver:
     [java] Server is established!
```

In this example, we will deploy 8 indexservers as the PA3 demands. Then we execute ant runserver for the other 7 indexservers.



```
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

```
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

```
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

```
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

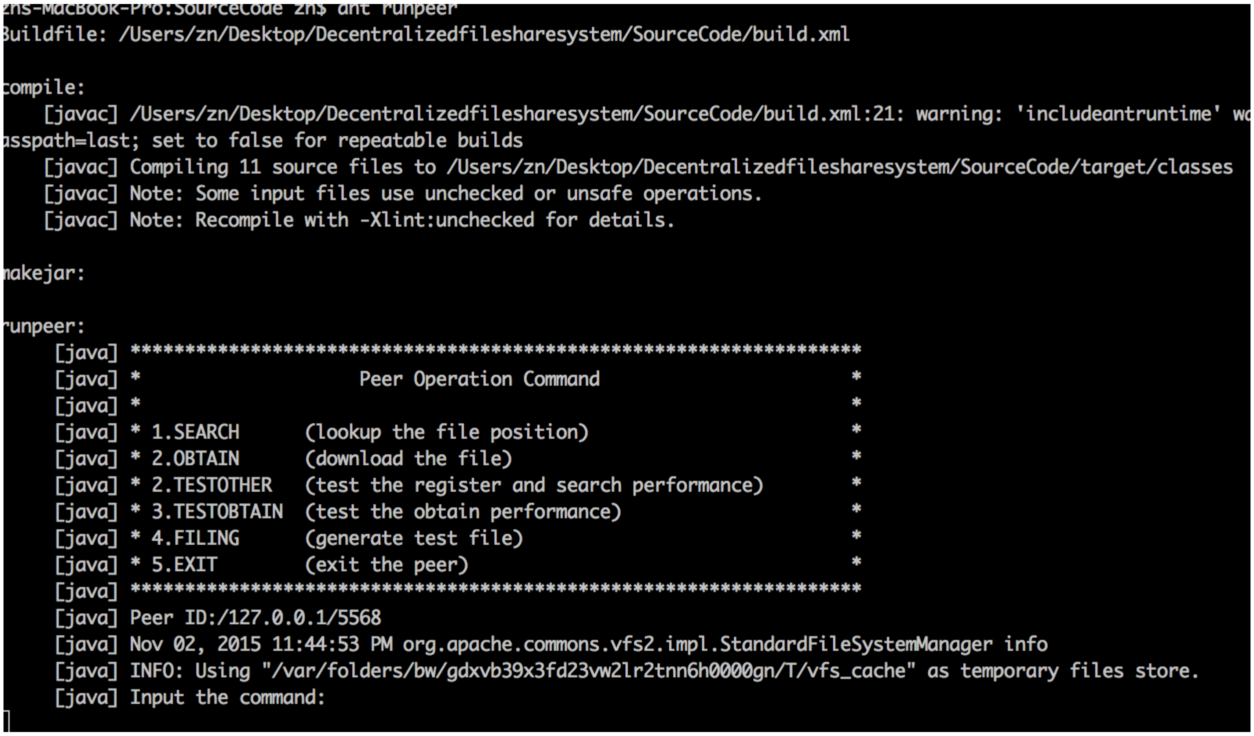
```
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

```
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

```
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

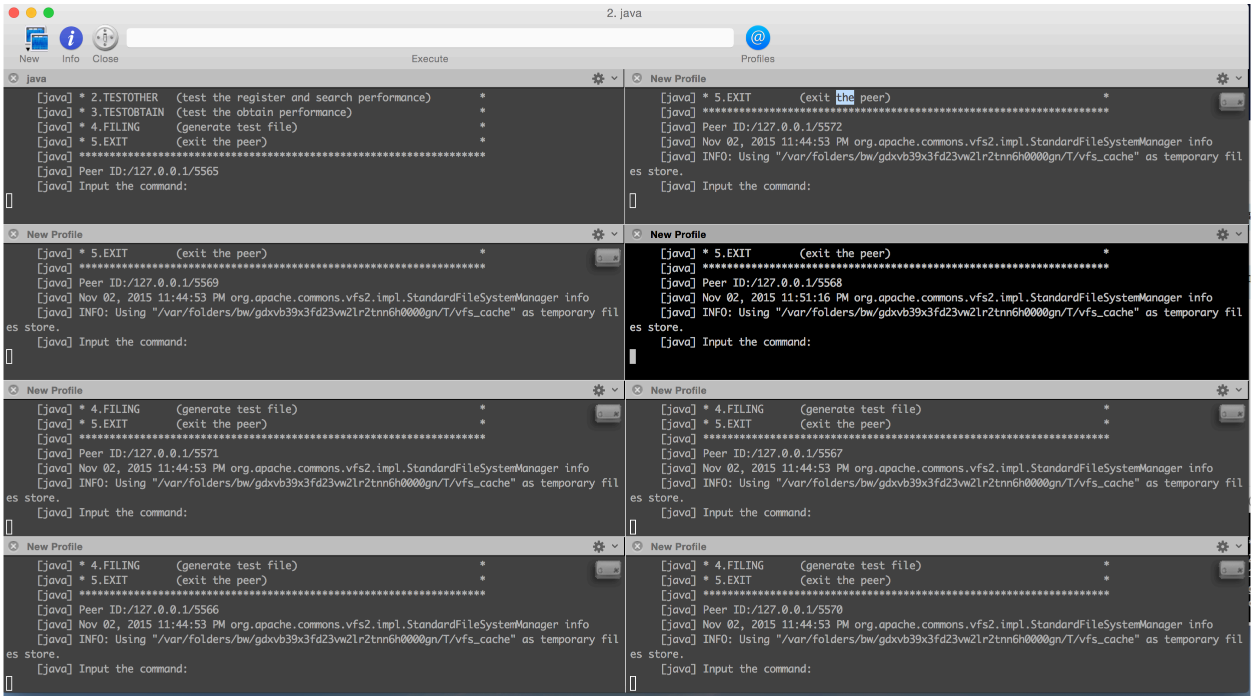
```
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

Step4. Then entering the command 'ant runpeer' in terminal to run the Peer class which is the client and fileserver. We can see the outputs in the following screenshot.



```
zns-MacBook-Pro:SourceCode zn$ ant runpeer
Buildfile: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/build.xml

compile:
    [javac] /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/build.xml:21: warning: 'includeantruntime' wa
asspath=last; set to false for repeatable builds
    [javac] Compiling 11 source files to /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/classes
    [javac] Note: Some input files use unchecked or unsafe operations.
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:

runpeer:
     [java] *******************************************************************
     [java] *                    Peer Operation Command                       *
     [java] *                                                                 *
     [java] * 1.SEARCH      (lookup the file position)                        *
     [java] * 2.OBTAIN      (download the file)                               *
     [java] * 2.TESTOTHER   (test the register and search performance)        *
     [java] * 3.TESTOBTAIN  (test the obtain performance)                     *
     [java] * 4.FILING      (generate test file)                              *
     [java] * 5.EXIT        (exit the peer)                                   *
     [java] *******************************************************************
     [java] Peer ID:/127.0.0.1/5568
     [java] Nov 02, 2015 11:44:53 PM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary files store.
     [java] Input the command:
```

In this example, we will deploy 8 clients as the PA3 demands. Then we execute ant runpeer for the other 7 clients.



java

```
     [java] * 2.TESTOTHER   (test the register and search performance)        *
     [java] * 3.TESTOBTAIN  (test the obtain performance)                     *
     [java] * 4.FILING      (generate test file)                              *
     [java] * 5.EXIT        (exit the peer)                                   *
     [java] *******************************************************************
     [java] Peer ID:/127.0.0.1/5565
     [java] Input the command:
```

New Profile

```
     [java] * 5.EXIT        (exit the peer)                                   *
     [java] *******************************************************************
     [java] Peer ID:/127.0.0.1/5572
     [java] Nov 02, 2015 11:44:53 PM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fil
es store.
     [java] Input the command:
```

New Profile

```
     [java] * 5.EXIT        (exit the peer)                                   *
     [java] *******************************************************************
     [java] Peer ID:/127.0.0.1/5569
     [java] Nov 02, 2015 11:44:53 PM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fil
es store.
     [java] Input the command:
```

New Profile

```
     [java] * 5.EXIT        (exit the peer)                                   *
     [java] *******************************************************************
     [java] Peer ID:/127.0.0.1/5568
     [java] Nov 02, 2015 11:51:16 PM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fil
es store.
     [java] Input the command:
```

New Profile

```
     [java] * 4.FILING      (generate test file)                              *
     [java] * 5.EXIT        (exit the peer)                                   *
     [java] *******************************************************************
     [java] Peer ID:/127.0.0.1/5571
     [java] Nov 02, 2015 11:44:53 PM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fil
es store.
     [java] Input the command:
```

New Profile

```
     [java] * 4.FILING      (generate test file)                              *
     [java] * 5.EXIT        (exit the peer)                                   *
     [java] *******************************************************************
     [java] Peer ID:/127.0.0.1/5567
     [java] Nov 02, 2015 11:44:53 PM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fil
es store.
     [java] Input the command:
```

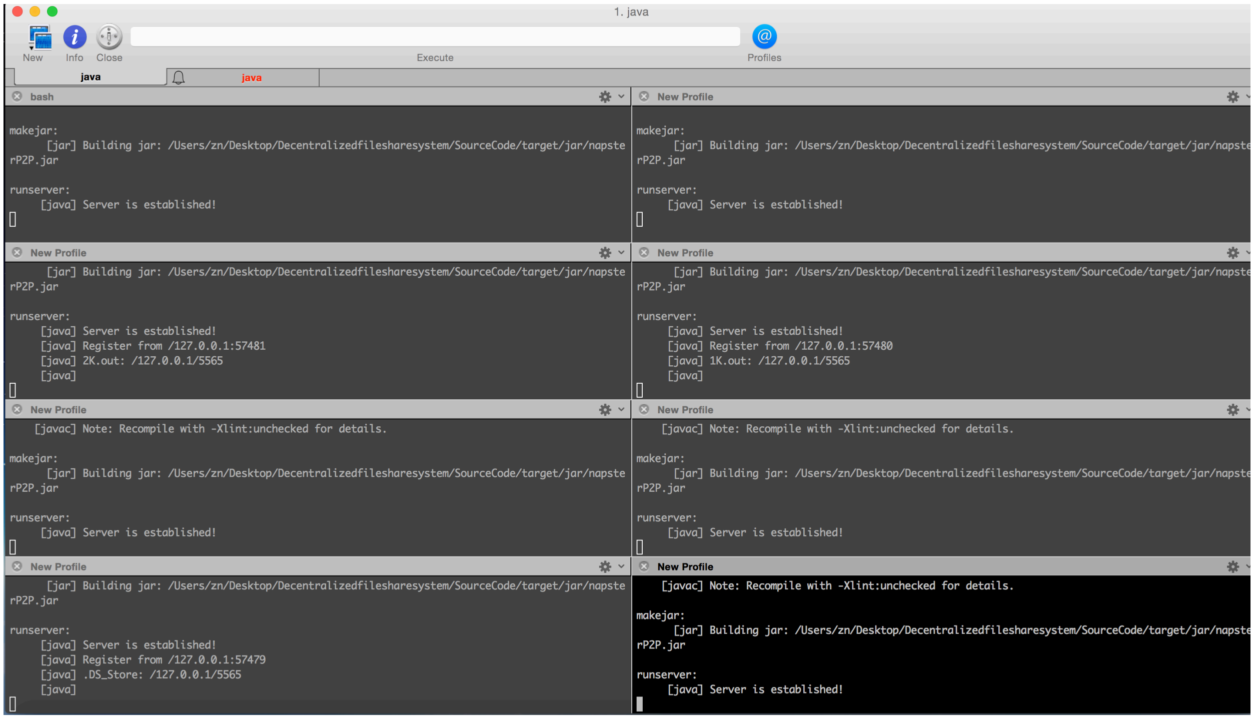
New Profile

```
     [java] * 4.FILING      (generate test file)                              *
     [java] * 5.EXIT        (exit the peer)                                   *
     [java] *******************************************************************
     [java] Peer ID:/127.0.0.1/5566
     [java] Nov 02, 2015 11:44:53 PM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fil
es store.
     [java] Input the command:
```

New Profile

```
     [java] * 4.FILING      (generate test file)                              *
     [java] * 5.EXIT        (exit the peer)                                   *
     [java] *******************************************************************
     [java] Peer ID:/127.0.0.1/5570
     [java] Nov 02, 2015 11:44:53 PM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fil
es store.
     [java] Input the command:
```

Step5.we copy several files into the filesever directory, then the client will automatically register to the indexserver. We can see the outputs in the following screenshot.



```
makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

```
makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

```
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
     [java] Register from /127.0.0.1:57481
     [java] 2K.out: /127.0.0.1/5565
     [java]
```

```
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
     [java] Register from /127.0.0.1:57480
     [java] 1K.out: /127.0.0.1/5565
     [java]
```

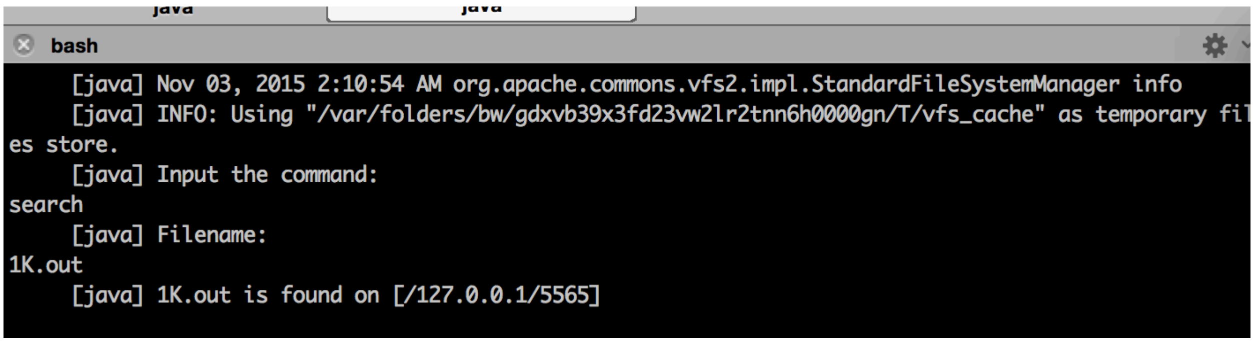
```
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

```
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

```
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
     [java] Register from /127.0.0.1:57479
     [java] .DS_Store: /127.0.0.1/5565
     [java]
```

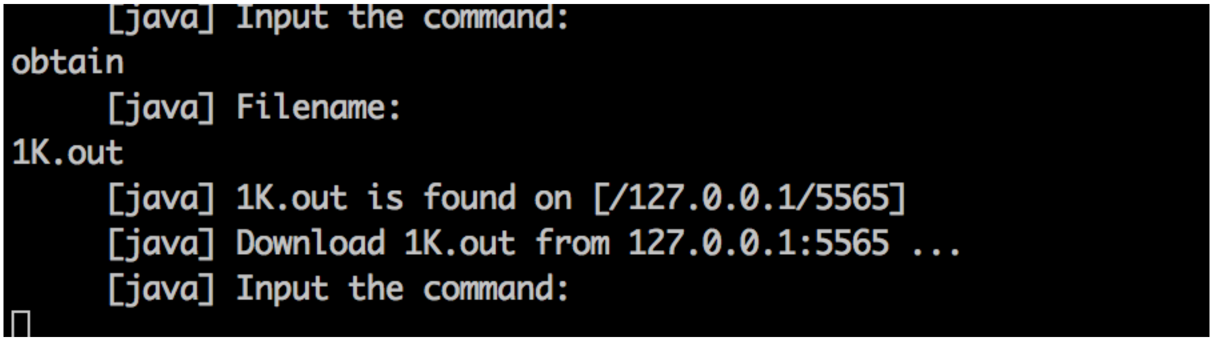
```
    [javac] Note: Recompile with -Xlint:unchecked for details.

makejar:
      [jar] Building jar: /Users/zn/Desktop/Decentralizedfilesharesystem/SourceCode/target/jar/napste
rP2P.jar

runserver:
     [java] Server is established!
```

Step6. We search the files which we just put into the fileserver directory. We can see the outputs in the following screenshot.



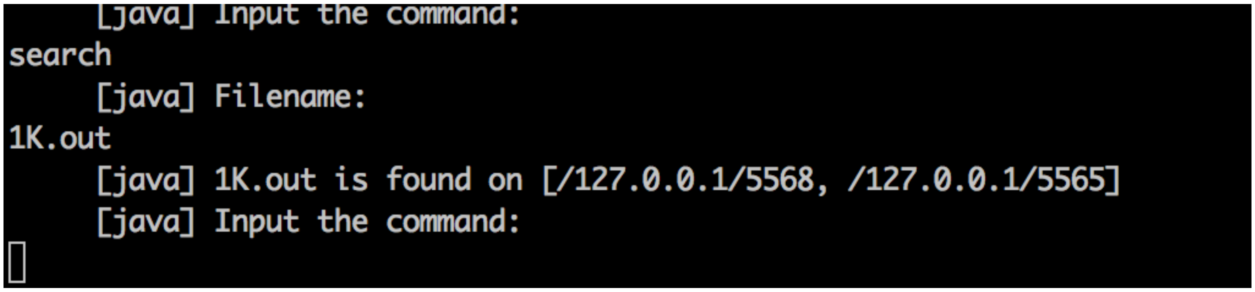
```
     [java] Nov 03, 2015 2:10:54 AM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fil
es store.
     [java] Input the command:
search
     [java] Filename:
1K.out
     [java] 1K.out is found on [/127.0.0.1/5565]
```

Step7. We obtain the files which we just put into the fileserver directory. We can see the outputs in the following screenshot.



```
     [java] Input the command:
obtain
     [java] Filename:
1K.out
     [java] 1K.out is found on [/127.0.0.1/5565]
     [java] Download 1K.out from 127.0.0.1:5565 ...
     [java] Input the command:
```

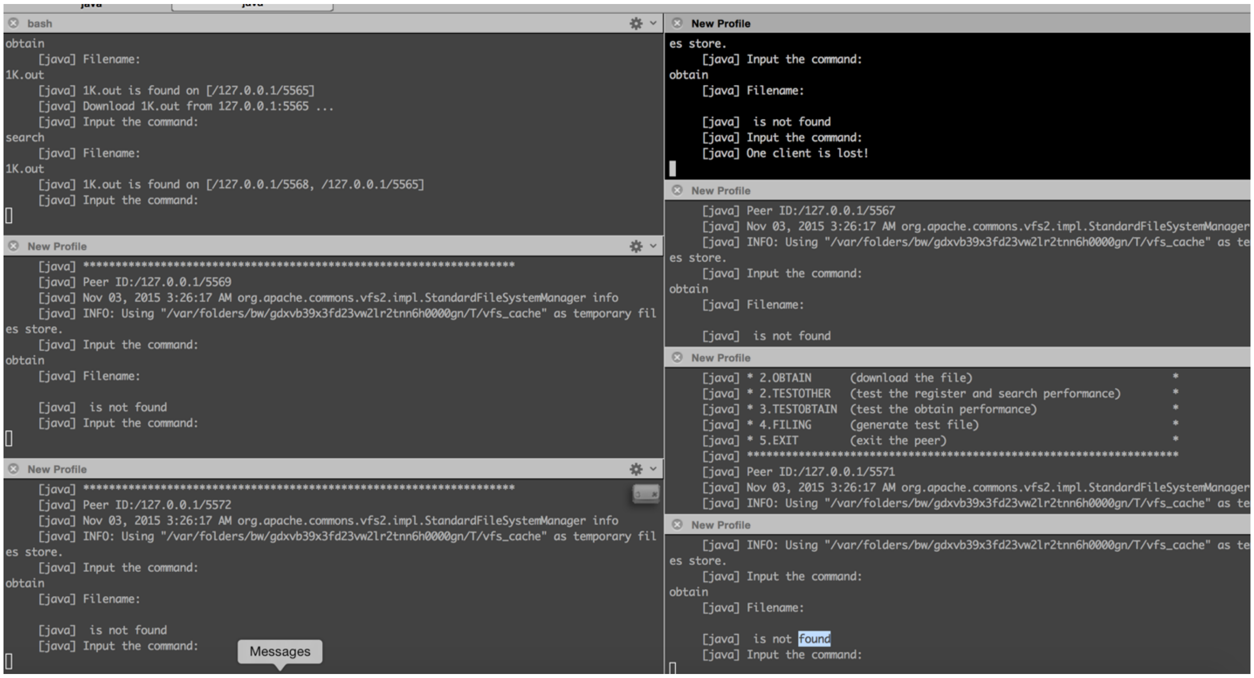
Step8. In order to prove that the file is obtained successfully, we search this file to lookup the location of this file.



```
     [java] Input the command:
search
     [java] Filename:
1K.out
     [java] 1K.out is found on [/127.0.0.1/5568, /127.0.0.1/5565]
     [java] Input the command:
```

we can see the file 1K.out is in the fileserver /127.0.0.1/5568. It prove this fileserver obtain this file successfully.

Step9. In order to prove that the file replication is working, we just close a fileserver whose ID is /127.0.0.1/5565. And if we obtain the file 2K.out successfully, it proves that the file replication is working. Because the file 2K.out is registered from the fileserver /127.0.0.1/5565, which means this file is not in other fileservers except the fileserver /127.0.0.1/5565. When we close this fileserver, we can get the outputs.



```
obtain
     [java] Filename:
1K.out
     [java] 1K.out is found on [/127.0.0.1/5565]
     [java] Download 1K.out from 127.0.0.1:5565 ...
     [java] Input the command:
search
     [java] Filename:
1K.out
     [java] 1K.out is found on [/127.0.0.1/5568, /127.0.0.1/5565]
     [java] Input the command:
```

```
     [java] ******************************************************************
     [java] Peer ID:/127.0.0.1/5569
     [java] Nov 03, 2015 3:26:17 AM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fil
es store.
     [java] Input the command:
obtain
     [java] Filename:

     [java]  is not found
     [java] Input the command:
```

```
     [java] ******************************************************************
     [java] Peer ID:/127.0.0.1/5572
     [java] Nov 03, 2015 3:26:17 AM org.apache.commons.vfs2.impl.StandardFileSystemManager info
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as temporary fil
es store.
     [java] Input the command:
obtain
     [java] Filename:

     [java]  is not found
     [java] Input the command:
```

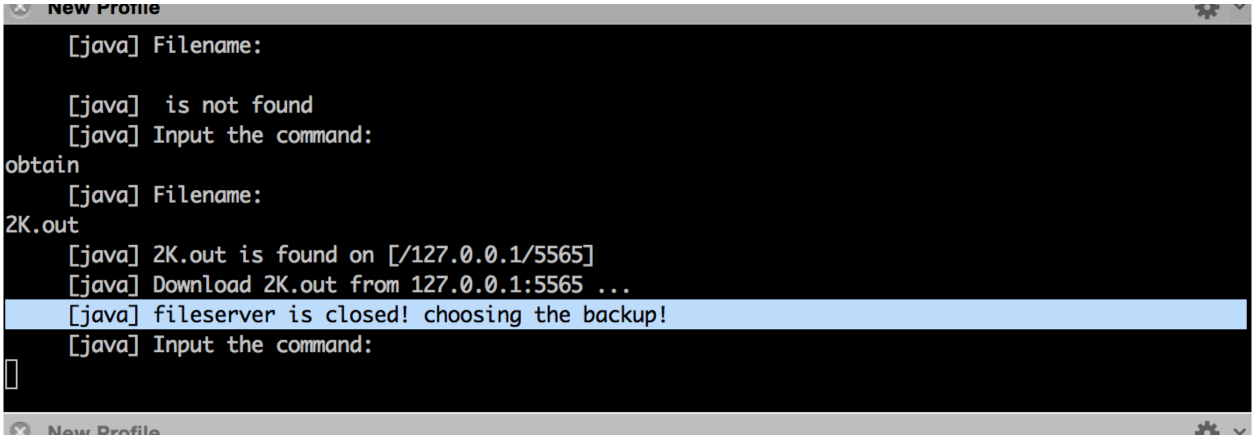
```
es store.
     [java] Input the command:
obtain
     [java] Filename:

     [java]  is not found
     [java] Input the command:
     [java] One client is lost!
```

```
     [java] Peer ID:/127.0.0.1/5567
     [java] Nov 03, 2015 3:26:17 AM org.apache.commons.vfs2.impl.StandardFileSystemManager
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as te
es store.
     [java] Input the command:
obtain
     [java] Filename:

     [java]  is not found
```

```
     [java] * 2.OBTAIN      (download the file)                                    *
     [java] * 2.TESTOTHER   (test the register and search performance)             *
     [java] * 3.TESTOBTAIN  (test the obtain performance)                          *
     [java] * 4.FILING      (generate test file)                                   *
     [java] * 5.EXIT        (exit the peer)                                        *
     [java] ******************************************************************
     [java] Peer ID:/127.0.0.1/5571
     [java] Nov 03, 2015 3:26:17 AM org.apache.commons.vfs2.impl.StandardFileSystemManager
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as te
```

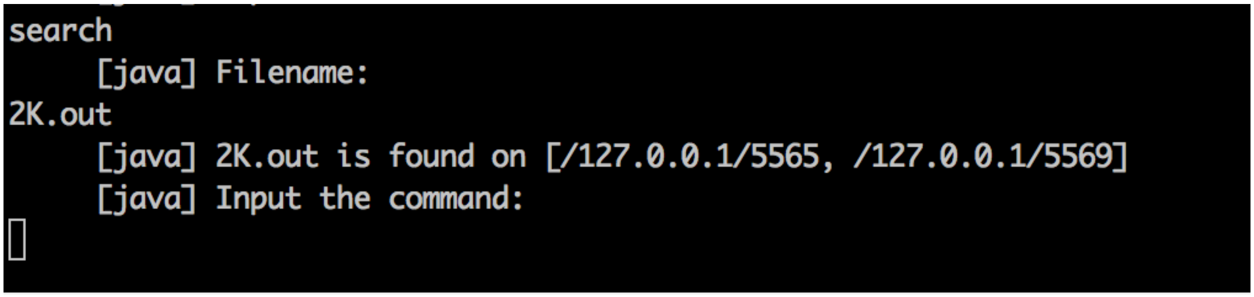
```
     [java] INFO: Using "/var/folders/bw/gdxvb39x3fd23vw2lr2tnn6h0000gn/T/vfs_cache" as te
es store.
     [java] Input the command:
obtain
     [java] Filename:

     [java]  is not found
     [java] Input the command:
```

Then we obtain the file 2K.out in any client. We can get the outputs as following.



```
     [java] Filename:

     [java]  is not found
     [java] Input the command:
obtain
     [java] Filename:
2K.out
     [java] 2K.out is found on [/127.0.0.1/5565]
     [java] Download 2K.out from 127.0.0.1:5565 ...
     [java] fileserver is closed! choosing the backup!
     [java] Input the command:
```

Then we search this file to prove that it obtains this file successfully.



```
search
     [java] Filename:
2K.out
     [java] 2K.out is found on [/127.0.0.1/5565, /127.0.0.1/5569]
     [java] Input the command:
```

this file is in the fileserver /127.0.0.1/5569 which obtains this file. It proves that the fileserver /127.0.0.1/5569 obtains this file successfully. And it also proves that our replication for files works well.