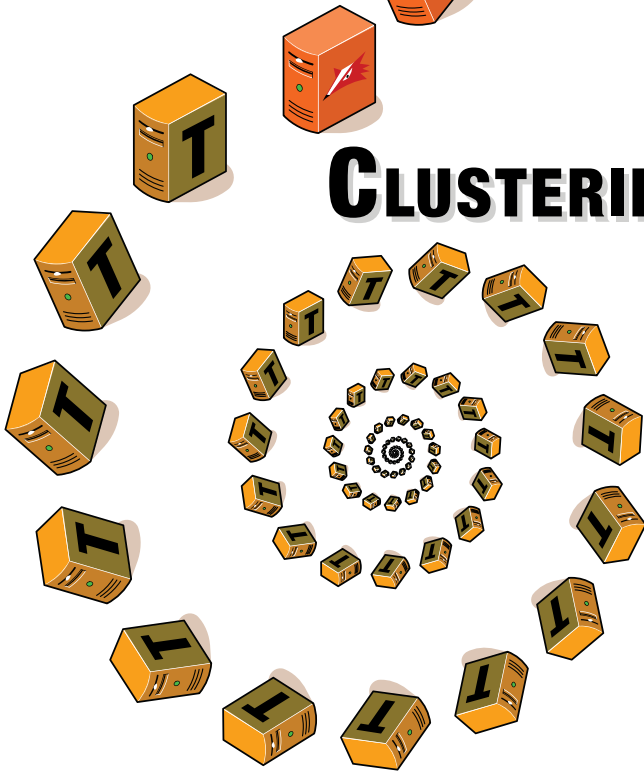


CLUSTERING SMARTFOXSERVER USING TERRACOTTA



DOCUMENTATION CREATED BY: **AS1 INTEGRATED**

107 ATLANTIC AVE.

TORONTO ONT.

M6K 1Y2 CANADA

AS1INTEGRATED.COM

416.703.2300



CLUSTERING SMARTFOXSERVER USING TERRACOTTA

Table of Contents

- 1. Preparing the Hardware** *pg.3*
- 2. Preparing the Operating System** *pg.4-5*
- 3. Preparing the Software** *pg.5*
- 4. Software Configuration** *pg.6-8*
- 5. Starting up the Cluster** *pg.9-10*
- 6. Monitoring the Services** *pg.11-14*
- 7. FAQ** *pg.15*

CLUSTERING SMARTFOXSERVER USING TERRACOTTA

Section 1- Preparing the Hardware

To create a base level fully redundant SmartFoxServer cluster, we will need a minimum of four(4) servers & one load balancer (a database would be optional). Terracotta will be setup in a 'PASSIVE/ACTIVE' setup over two(2) machines and the Smartfox server will be setup over two(2) machines as well. The load balancer will be setup to distribute the incoming Smartfox Server connections between SFS(Smartfox Server) #1 and SFS #2. See 'fig.1'.

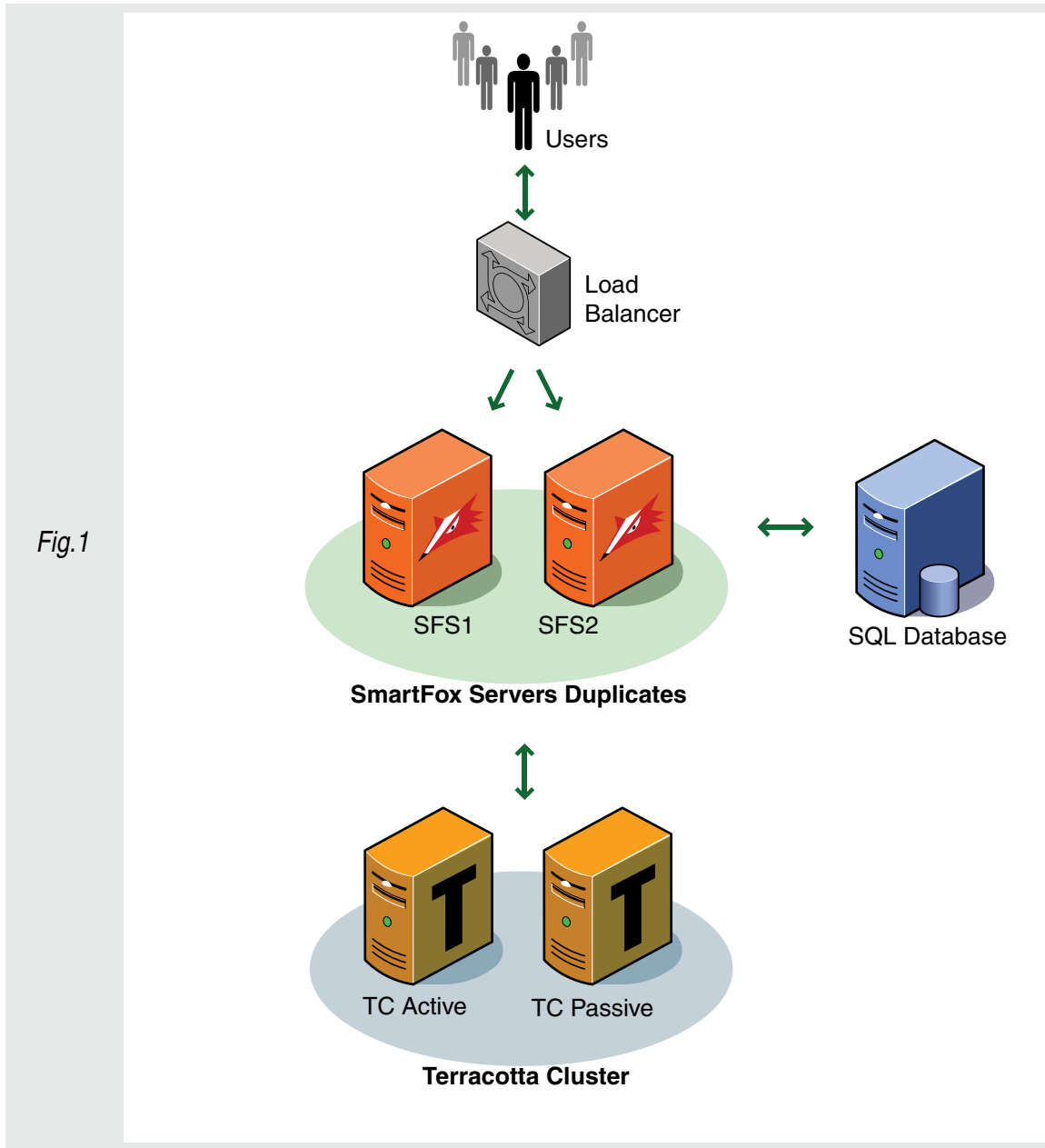


Fig.1

Note, that the SQL server is optional, but will be used in our examples to keep the users Buddy List and other variables set across both SFS. Also, The load balancer is used to equally place each new connection across the two SFS. This white-paper will skip the load balancer setup, as any software or hardware load balancer can be used to complete this simple action. By default the load balancer will need to forward connections on ports 9339 and 8080 for Smartfox to establish the socket connection.

CLUSTERING SMARTFOXSERVER USING TERRACOTTA

Section 2- Preparing the Operating System

In these examples we will be using CentOS version 5, but keep in mind that any Linux distribution can be used with this documented cluster configuration. To keep the confusion down to a minimum, we will install a base setup of the OS(Operating System) on each of the cluster machines. For CentOS-5 we can follow these step by step actions.

1. Boot with CD #1 in the drive (and BIOS set to boot from CD-ROM).
2. Press 'Enter' When prompted to enter a graphical install setup.
3. Skip the CD test.
4. When the graphical install setup screen appears, click 'Next'
5. Select language - English (English) - click 'Next'
6. Select keyboard layout - U.S. English - click 'Next'
7. Select the drop down menu - Select 'Remove all partitions on selected drives and create default layout' - Check the 'Review and modify partitioning layout' check box - click 'Next'
8. Click 'Yes' when asked if you are sure to remove all partitions.
9. Review the partition layout - click 'Next'
10. Select 'GRUB' for the boot loader - click 'Next'
11. Select eth0 and have it setup for automatic DHCP - click 'Next'
12. Set the timezone (America/Toronto) - Check the use 'UTC' check box - click 'Next'
13. Enter a root password (Document it!) - click 'Next'
14. Uncheck all boxes from the software list.
15. Select 'Customize No' radio button - click 'Next'
16. Should be ready to start installing the operating system now - click 'Next' to begin
17. Get the CD's ready that it prompts you for. (CD #1, #2, #3, #4) -click 'Continue'
18. When prompted put in the other CD's - click 'OK'
19. When finished click 'Reboot'
20. On the first boot up of the machine, the setup agent will appear, Select 'Exit'.

Be sure to set a good range of IP addresses for the new servers. For this documentation we will use a internal set of IP's starting with 50. So our cluster IP setup should look something like the following.

192.168.1.50 - Load Balancer
192.168.1.51 - Terracotta Server 1
192.168.1.52 - Terracotta Server 2
192.168.1.53 - Smartfox Server 1
192.168.1.54 - Smartfox Server 1
192.168.1.55 - SQL Server

CLUSTERING SMARTFOX SERVER USING TERRACOTTA

Section 2- Preparing the Operating System (cont...)

Finally, be sure to get every machine updated with the latest patches and packages. In this documentation we will be running all of our shell commands as root. If you are not logged in as root be sure you're a sudo user and 'sudo' each of these commands. In CentOS, yum will take care of all of our package management.

```
[root@A51-S01 ~]# yum update
```

yum will then retrieve and install all updated packages for the CentOS install. Finally, there is one extra package we will need on each server and that is 'screen' will get more into screen later, but for now install it with the command:

```
[root@A51-S01 ~]# yum install screen
```

Finally, we will need to configure iptables to allow the machines to talk to each other. In this documentation we will take the easy route and just disable the iptables service, but in a production environment, it should be setup with a specific set of rules. To shutdown the server and make sure it won't start again on reboot run the following commands:

```
[root@A51-S01 ~]# /etc/init.d/iptables stop && chmod -x /etc/init.d/iptables
```

Once this is done to each machine in the cluster, we are ready to start setting up the specific software on each of them. Depending on how the yum updates went (ie. a new kernel was retrieved - you may want to restart the server. Best practice is to just do this anyways).

Section 3- Preparing the Software

For this white paper we will be using a sample package (clusterar.gz) which includes Smartfox Server, Terracotta and a JRE which can be downloaded from the Smartfox website. This package contains all required software to complete this cluster. The archive will then be placed on each machine in the cluster and unarchived from there.

SFTP into each machine and load clusterar.gz into the /root directory. Follow that action by remoting into the machine and extracting the files from that file. Use the following commands:

```
[root@A51-S01 ~]# tar xzvf cluster.tar.gz
```

We should now have a new directory in /root called 'cluster'. We can go ahead and remove the tar.gz file and also rename the new directory to reflect the machine we are on:

```
[root@A51-S01 ~]# rm cluster.tar.gz && mv cluster tc1
```

**Note: In the previous shell command I moved the folder to a new folder called tc1(Terracotta #1) - My two Terracotta server will have tc1 & tc2 and my two Smartfox Servers will have sfs1 & sfs2.*

Now we will start a new screen session to manage the rest of the server in.

```
[root@A51-S01 ~]# screen -S tc1
```

**Note: Again I have used tc1 in this command, please reflect the screen name properly for every server you're working on.*

Once we have loaded & extracted the files on each server as well as started a screen session we can move on to the next step.

CLUSTERING SMARTFOXSERVER USING TERRACOTTA

Section 4- Software Configuration

Now we will start to get into the tricky bit of things. Each server will need to be configured in the proper manner so that they may all work together without any issues. I will break this section down into two parts, the Terracotta config, and the Smartfox Server config. We will start with setting up the Terracotta config.

Terracotta Config

As I stated before, we are going to setup TC(Terracotta) in a 'ACTIVE/PASSIVE' manner. This means that all the workload will be done by the ACTIVE server while it is running. If the ACTIVE server should for any reason die, the connections that are currently handled by the ACTIVE server will then be passed on to the PASSIVE server as well as making the PASSIVE server the new ACTIVE server. Once the old ACTIVE server is brought back up, it will see that the old PASSIVE server is a ACTIVE server and make itself the new PASSIVE server, ready for another ACTIVE server failure.

Lets start by working on tc1:

Make sure we are connected to tc1 and have the tc1 screen attached. To reattach a screen session use the command:

```
[root@A51-S01 ~]# screen -r tc1
```

Start by moving into the 'bin' directory of the tc application folder:

```
[root@A51-S01 ~]# cd /root/tc1/terraccotta-2.5.2/bin && ll
```

This is the directory where we will do all of our work on TC. As you can see from the directory listing, we have a start and stop script ready for our use.

```
-rwxr-xr-x 1 1000 1000 979 Feb 18 2008 admin.sh
-rwxr-xr-x 1 1000 1000 1003 Feb 18 2008 archive-tool.sh
-rwxr-xr-x 1 1000 1000 2303 Feb 18 2008 boot-jar-path.sh
-rwxr-xr-x 1 1000 1000 1721 Feb 18 2008 dso-env.sh
-rwxr-xr-x 1 1000 1000 677 Feb 18 2008 dso-java.sh
-rwxr-xr-x 1 1000 1000 1065 Mar 28 12:34 make-boot-jar.sh
-rwxr-xr-x 1 1000 1000 1013 Feb 18 2008 run-dgc.sh
-rwxr-xr-x 1 1000 1000 1025 Feb 18 2008 scan-boot-jar.sh
-rwxr-xr-x 1 1000 1000 1103 Mar 28 12:34 start-tc-server.sh
-rwxr-xr-x 1 1000 1000 1035 Mar 28 12:35 stop-tc-server.sh
-rwxr-xr-x 1 1000 1000 1000 Feb 18 2008 version.sh
```

But before we start up the service - we need to create a config file to let the server know what our intentions are. The easiest way to do this is to create a new file with a text editor - I will use nano.

```
[root@A51-S01 ~]# nano tc-config.xml
```

CLUSTERING SMARTFOXSERVER USING TERRACOTTA

Section 4- Software Configuration

(cont...)

Inside that file create a XML structure somewhat similar to the following (making changes to the IP addresses that you have used).

```
<?xml version="1.0" encoding="UTF-8"?>

<tc:tc-config xmlns:tc="http://www.terracotta.org/config"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.terracotta.org/schema/terracotta-4.xsd">

  <servers>

    <server host="192.168.1.51" name="tc1">
      <data>/root/terracotta/tc1-data</data>
      <logs>/root/terracotta/tc1-logs</logs>
      <l2-group-port>9530</l2-group-port>
    </server>

    <server host="192.168.1.52" name="tc2"> <data>/root/terracotta/tc2-data</data>
      <logs>/root/terracotta/tc2-logs</logs>
      <l2-group-port>9530</l2-group-port>
    </server>

    <ha>
      <mode>networked-active-passive</mode>
      <networked-active-passive>
      <election-time>5</election-time>
      </networked-active-passive>
    </ha>

  </servers>

</tc:tc-config>
```

Again be sure to reflect the proper IP & Port addresses as well as the log directory you wish to have. Be sure to leave the server name the same as listed above as they will need to be used in some commands in the near future.

We can now do the exact same setup to tc2. Be sure to name the screen session properly (ie. screen -S tc2). The XML structure can stay the same on both TC servers because our main concern is what the server name is that we have listed in the XML doc. The fastest way to get the XML document over to tc2 is to probably just SCP it.

```
[root@A51-S01 ~]# scp tc-config.xml 192.168.1.52:~/root/tc1/terracotta-2.5.2/bin/
```

It's just that easy! For now let's just leave these servers and go on to setup the configuration on the two Smartfox Servers.

CLUSTERING SMARTFOX SERVER USING TERRACOTTA

Section 4- Software Configuration

(cont...)

Smartfox Server Config

The two Smartfox Servers will need to be setup exactly the same (except a couple IP address declarations) so everything I do in the following should be reflected on both sfs1 and sfs2. Start by remoting into the SFS and reattaching the screen if its not already attached.

```
[root@A51-S01 ~]# screen -r sfs1
```

Next, move yourself into the SFS application directory:

```
[root@A51-S01 ~]# cd /root/sfs1/Server && ll
```

There is a couple config variables we will need to change in the following source. The first one being in the 'tc-cluster-lobby.xml' file. Open the file in a editor, again I will use nano:

```
[root@A51-S01 ~]# nano tc-cluster-lobby.xml
```

You will see a file similar to the following - the only change needed for now is to point the IP address to the IP of your ACTIVE Terracotta server.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
    All content copyright (c) 2003-2007 Terracotta, Inc.,
    except as may otherwise be noted in a separate copyright notice.
    All rights reserved.
-->
<tc:tc-config xmlns:tc="http://www.terracotta.org/config"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.terracotta.org/schema/terracotta-4.xsd">
  <!-- Server is colocated with client and DSO is listening on
    default port (9510). -->
  <servers>
    <server host="192.168.1.51" name="tc1" />
    <update-check>
      <enabled>>false</enabled>
    </update-check>
  </servers>

  <!-- Tell DSO where to put the generated client logs.
    See the Terracotta Configuration Guide and Reference for additional
    information. -->
  <clients>
    <logs>%(user.home)/terracotta/client-logs/pojo/cluster-lobby/%D</logs>
  </clients>

  <application>
    .... the file goes on....
```

Be sure to update the

```
<server host="192.168.1.51" name="tc1" />
```

line to reflect the proper IP address. Our second config change is with the XML config document for Smartfox itself. The best way to do this is to either start from scratch with the 'config_template.xml' file or use your own current SFS config file that you already have running from a previous server. That's it, Smartfox should be ready for you to start it up. But don't start it up yet - that's the next step!

CLUSTERING SMARTFOXSERVER USING TERRACOTTA

Section 5- Starting Up the Cluster

Ok it is finally time to see if all of your setup & configuration was done correctly. In order to get the cluster up and running we will start the services in order, start with Terracotta and tc1. Reattach the screen on tc1 if not already attached and run the following command.

```
[root@A51-S01 ~]# ./start-tc-server.sh -f tc-config.xml -n tc1
```

(*-f* indicates the config location, *-n* indicates the name of the server)

You should see the output of Terracotta starting up. Something like the following:

```
2008-07-25 11:19:56,588 INFO - Terracotta 2.5.2, as of 20080218-120204 (Revision 7031 by cruise@rh4mo0 from 2.5)
2008-07-25 11:19:57,465 INFO - Configuration loaded from the file at '/root/cluster_demo/terraccotta-2.5.2/bin/tc-config.xml'.
2008-07-25 11:19:57,545 INFO - Log file: '/root/terraccotta/demo-server/server1-logs/terraccotta-server.log'.
2008-07-25 11:19:58,305 INFO - JMX Server started. Available at URL[service:jmx:jmxmp://0.0.0.0:9520]
Jul 25, 2008 11:19:59 AM org.apache.catalina.tribes.transport.ReceiverBase bind
INFO: Receiver Server Socket bound to:/0.0.0.0:9530
2008-07-25 11:20:04,488 INFO - Becoming State[ ACTIVE-COORDINATOR ]
2008-07-25 11:20:04,558 INFO - Terracotta Server has started up as ACTIVE node on 0:0:0:0:0:0:9510 successfully, and is now ready for work.
```

If you get an error please read the FAQ.

Once the server is ready for work we can go ahead close the screen session (it will allow the application to continue to run without us being logged in). To close the screen session hit CTRL+A and then D. Now we can connect to tc2 and run the same command just substituting the *-n tc1* for *-n tc2*. It would look something like this:

```
[root@A51-S01 ~]# ./start-tc-server.sh -f tc-config.xml -n tc2
```

CLUSTERING SMARTFOXSERVER USING TERRACOTTA

Section 5- Starting Up the Cluster (cont...)

Running that command should give an output something close to the following:

```
2008-08-21 05:20:11,204 INFO - Terracotta 2.5.2, as of 20080218-120204 (Revision 7031 by cruise@rh4mo0 from 2.5)
2008-08-21 05:20:12,001 INFO - Configuration loaded from the file at '/root/tc2/terracotta-2.5.2/bin/tc-
config.xml'.
2008-08-21 05:20:12,065 INFO - Log file: '/root/terracotta/tc2-logs/terracotta-server.log'.
2008-08-21 05:20:12,799 INFO - JMX Server started. Available at URL[service:jmx:jmxmp://0.0.0.0:9520]
Aug 21, 2008 5:20:13 AM org.apache.catalina.tribes.transport.ReceiverBase bind
INFO: Receiver Server Socket bound to:/0.0.0.0:9530
2008-08-21 05:20:13,996 INFO - NodeID[tcp://192.168.1.51:9530] joined the cluster
Aug 21, 2008 5:20:14 AM org.apache.catalina.tribes.io.BufferPool getBufferPool
INFO: Created a buffer pool with max size:104857600 bytes of type:org.apache.catalina.tribes.io.BufferPool15Impl
2008-08-21 05:20:14,450 INFO - Moved to State[ PASSIVE-UNINITIALIZED ]
2008-08-21 05:20:14,527 INFO - Moved to State[ PASSIVE-STANDBY ]
```

Note the last line of output stating that the state is moved to '[PASSIVE-STANDBY]'. This is exactly what we want to see, as it shows that tc2 saw tc1 running and automatically placed itself in PASSIVE mode.

We can now move forward and get Smartfox up and running and talking to our new Terracotta cluster.

Getting Smartfox up and running should be even easier than getting the Terracotta service started. Connect to sfs1 and sfs2 and reattach the screen sessions. To get SFS running, we just need to run the start script with the SFS config XML document that we have created supplied.

```
[root@A51-S01 ~]# ./start-cluster.sh our_config.xml
```

The startup script will run and if everything starts up right - the final line of output should look something like the following:

```
05:24:43.671 - [ INFO ] > Server is up and running!
```

If you get an error please read the FAQ.
Everything should be up and running now!

CLUSTERING SMARTFOXSERVER USING TERRACOTTA

Section 6- Monitoring the Services

Terracotta can be easily monitored from the 'Terracotta Administration Console', simply have the application on your personal desktop machine and start it up. The main screen should look something like the following (Fig.2)

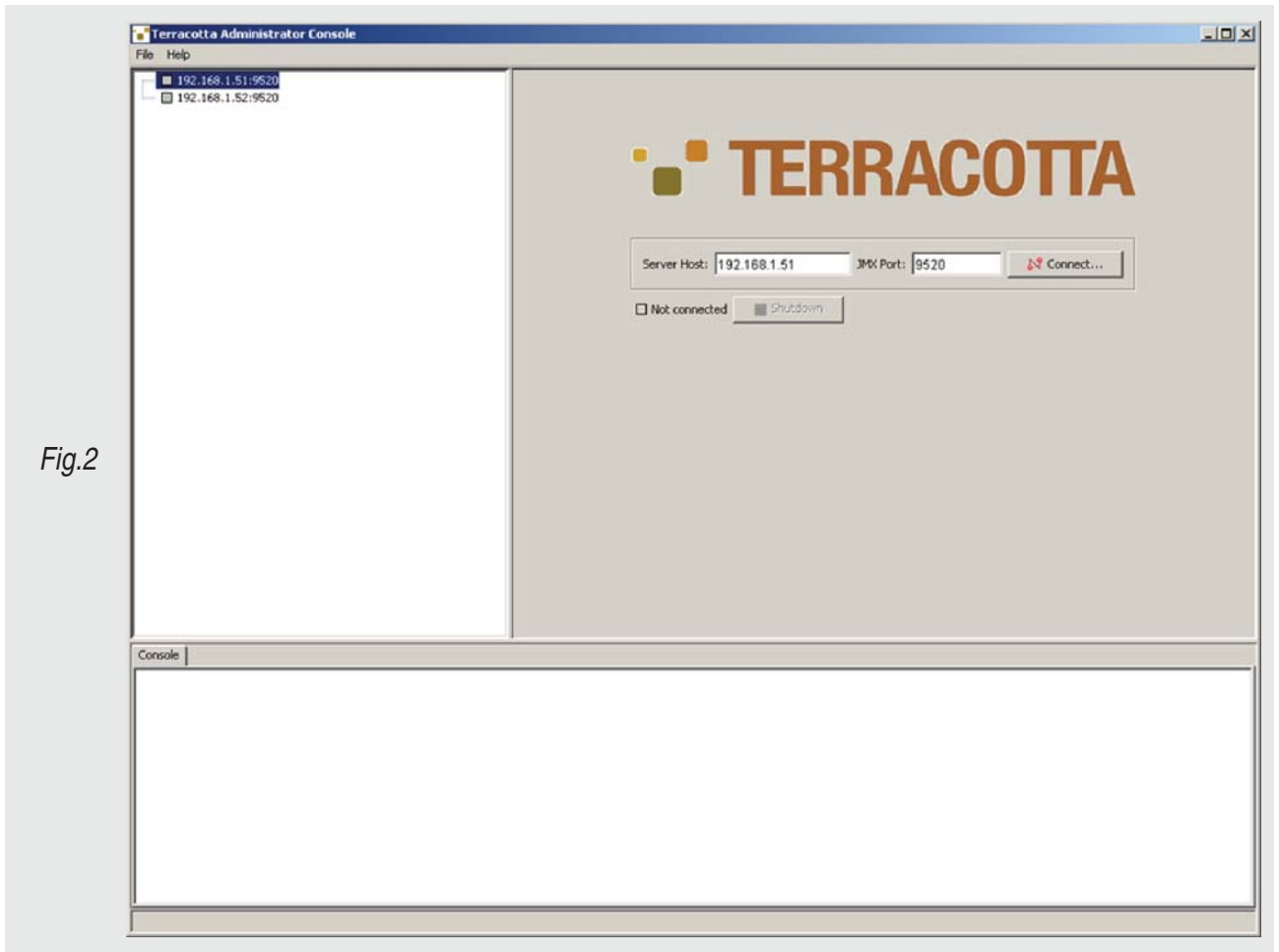


Fig.2

CLUSTERING SMARTFOXSERVER USING TERRACOTTA

Section 6- Monitoring the Services (cont...)

Enter the IP address of your active Terracotta server and try connecting to it. Your application should now look something like the following (Fig.3).
If you get an error please read the FAQ.

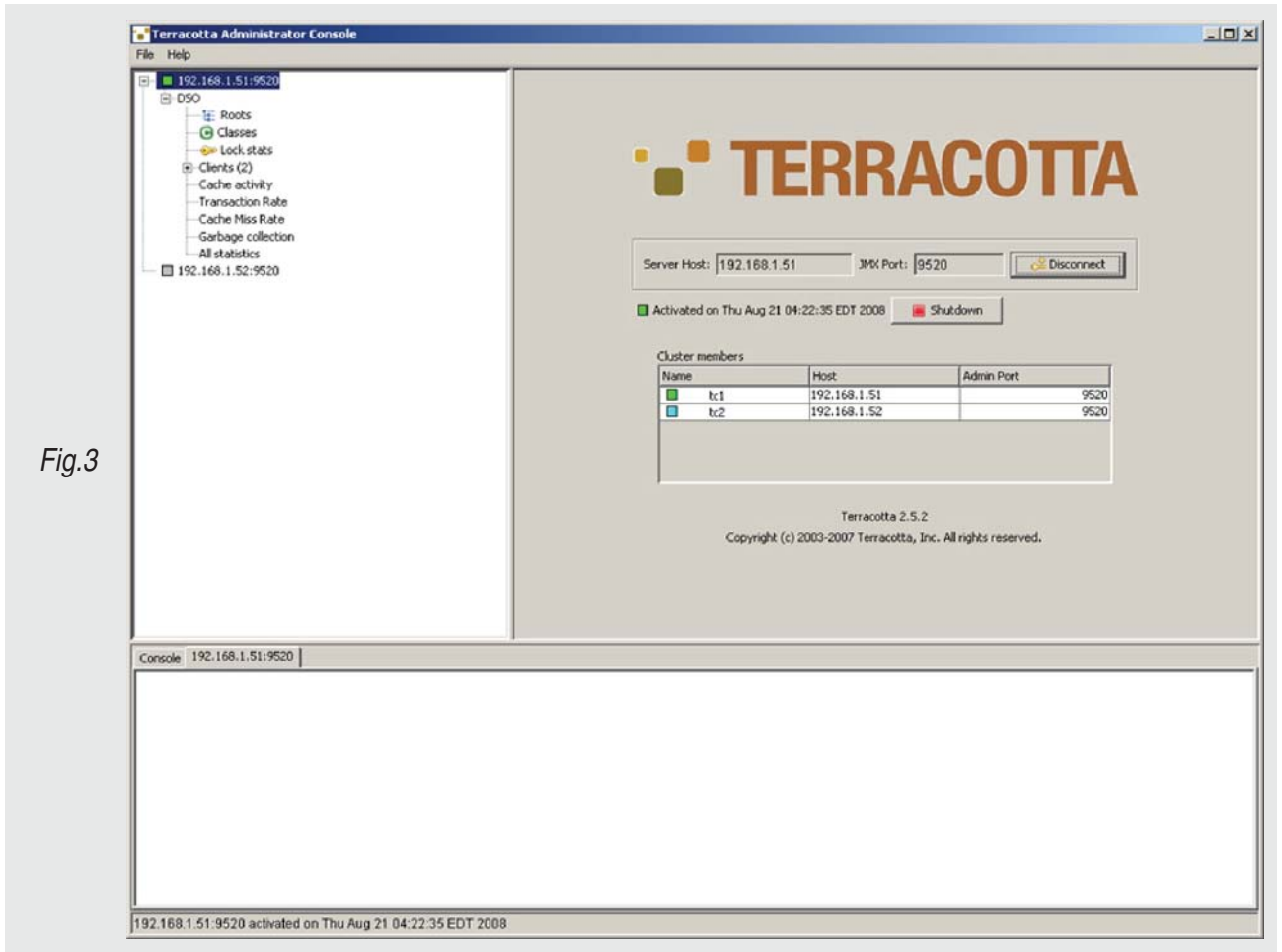


Fig.3

Notice that the 'Cluster Members' panel has two items in it. They should be our tc1 and tc2. tc1 should have a green square next to it show that is in ACTIVE mode and tc2 should be blue for PASSIVE mode. (Fig.4)

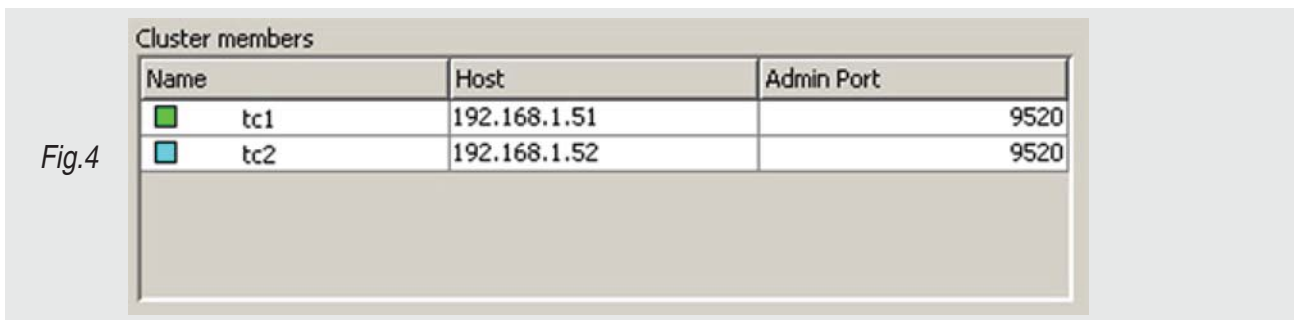


Fig.4

CLUSTERING SMARTFOXSERVER USING TERRACOTTA

Section 6- Monitoring the Services

(cont...)

The second thing we should notice is that the 'Clients' label under our tc1 DSO should have (2) beside it showing that both Smartfox Servers are connected. (Fig.5)

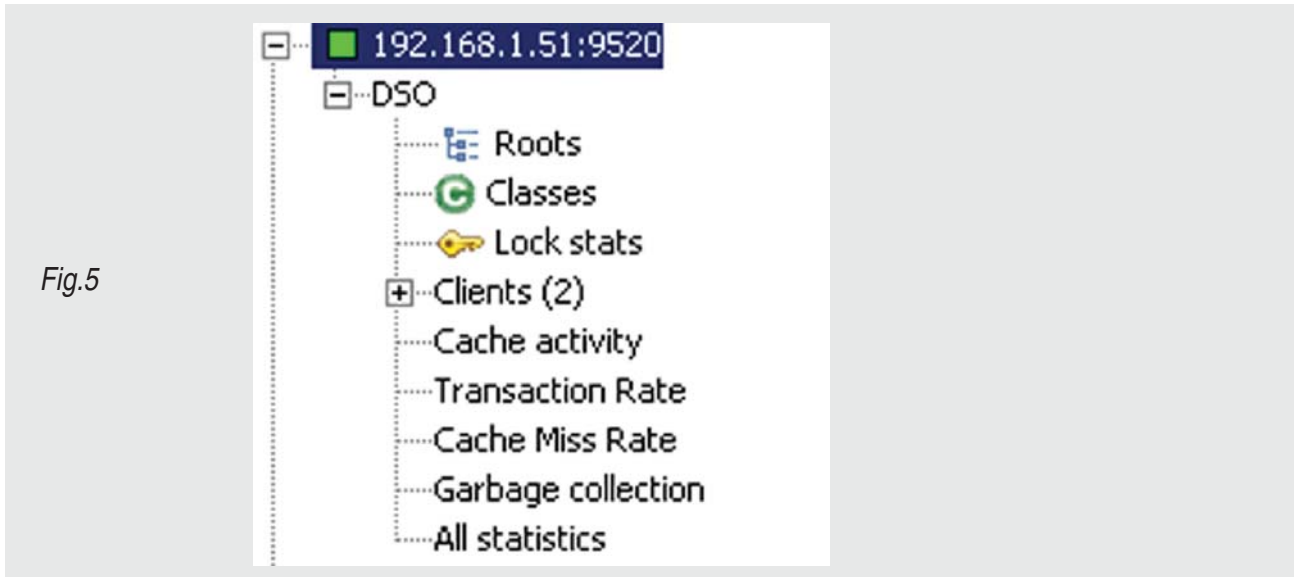


Fig.5

We can also check on our Smartfox servers in a similar fashion. Download and run the Smartfox AdminTool supplied by the Smartfox website. Enter the admin credentials set in your Smartfox config.xml file. (Fig.6)

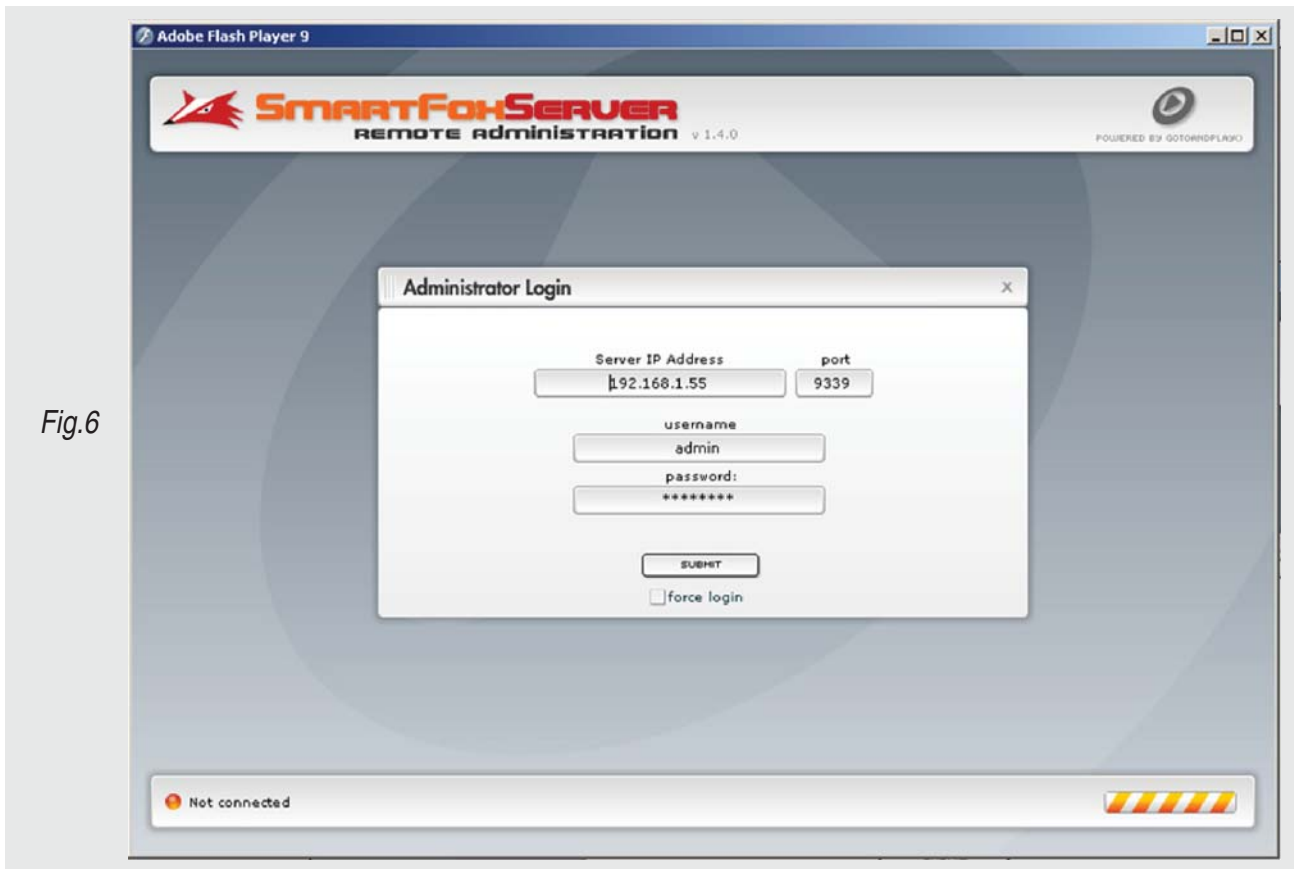


Fig.6

CLUSTERING SMARTFOXSERVER USING TERRACOTTA

Section 6- Monitoring the Services (cont...)

After logging in we should see a screen with the basic stats of our Smartfox Server. (Fig.7)



Fig.7

From here we can update many features of the Smartfox Server.
Like always, if you get an error please read the FAQ.

CLUSTERING SMARTFOXSERVER USING TERRACOTTA

Section 7- FAQ

Preparing the Hardware

Am I obligated to setup Terracotta in a ACTIVE/PASSIVE manner?

No, Terracotta is only setup in a ACTIVE/PASSIVE manner so that is one of the Terracotta servers happens to fail, the other one will take over. Without the PASSIVE server to tail behind the ACTIVE server, the Terracotta system is not redundant.

Why won't the Smartfox cluster share things like the Buddy List so that you don't have to use a external SQL server?

Smartfox Server comes packaged with a built in H2 database server that handles all of the data that a user needs to carry out their functions while connected to the Smartfox Server. Since the two Smartfox Servers each have their own H2 server, we have written a Smartfox Server side Extension to bypass the usage of H2 and push the data to a external SQL server.

Preparing the Operating System

Do I have to use CentOS to build this cluster?

No, not at all. We decided to use CentOS for our setup. Since there is no major requirements for the OS to have to run this cluster, you can get away with pretty much any OS you want including Windows and OS X.

Do I have to install just a base system for this cluster to run on?

No, but the less that is running on the server, the more horsepower to run this cluster.

Monitoring the Services

Where can I download the Smartfox Server Admin Tool?

You can always get the latest version of the Admin Tool by going to www.smartfoxserver.com and download the latest version of the Smartfox Server - inside the application directory there is a folder called 'Admin' and the Admin Tool is in there.