Steps to Setup a Multi-Tier Configuration

This is a rough outline on what it took me to setup a mutli-tier configuration for Barista/Addon. The clients are Windows (Vista and 7) and the server is CentOS 5.4, a free version of RedHat Linux.

Two Tier

1. Create fixed IP addresses for client and server. Put the names and IPs of the client and server into each hosts file (/etc/hosts on Linux, /Windows/system32/drivers/etc/hosts on Windows).
2. Install latest JDK on the server. Move JDK folder to /usr/local/. Change java link in /usr/bin/ to point at /usr/local/<jdk>. Set JAVA_HOME=/usr/local/<jdk> in the .bash_profile. (Setting up JAVA_HOME does not seem necessary if you set the java link.)
3. As root, install the latest BBj/Barista/Addon. I needed to execute it from the command line to get it started. (# java -jar *.jar) Start BBj Services automatically. ** On my CentOS system, I right-click on the .jar file and say Open With... => Java.
4. Allow port 8888 and 2103 through firewall on server and client.
5. Create a Linux user name for client user (e.g., ksnortum). Create an .rhosts in the home directory of ksnortum. Add a line with the client's system's name. Give it 644 permissions. Make ksnortum the owner. (Note: this step is considered unsafe by some people, but I'm not a network security person. Another option is to use User Authentication in the Environment tab of the EM. There is disagreement about whether running BBjSerives as a non-root user is a viable option) (On the Ubuntu distro, you cannot su to root. Use "sudo".)
6. At this point, all your Addon and Barista files are owned by "root". They need to be accessible by more than one user. Create a group called "barista" (or whatever you like). In aon/data and barista/sys/data change the group of all files to "barista". Change the permissions to 664.
7. The Linux users that will run Barista need to set a umask 0133. If all Linux users will be using Barista this can be done in /etc/bashrc or /etc/profile. Otherwise put the umask in each individual user's .bashrc or .profile in their home directory

Authentication with Enterprise Manager

As an alternative to using .rhosts you can use User Authentication in the Enterprise Manager (EM).

1. Follow steps one through four above.
2. You now have a choice whether to create Linux users. Let's start by not creating Linux users.
3. Go into the EM and turn on User Authentication (Server Information -> Environment tab).
4. Create EM users in the Users / Authentication area. It's advantageous, but not mandatory, to create EM users that are the same as your Barista users. Set passwords. Select User Permissions to Attach Existing DB and SQL Connections.
5. Create Barista users in Barista with the security roles etc. that you want. If they are the same name as your EM users, Barista will automatically log you in.
6. Create a group called "nobody". Change the group of all Addon and Barista data to "nobody". Give the data 664 permissions.

If you are creating Linux users too, do the following (why create Linux users? So they can log into the server and so you can have control over what owner and group files they get written with):

1. Do steps one through five above.
2. For each EM create a Linux user with the same name. Set a password. Create a group called "barista" (or whatever you wish). Assign this group to all Linux users that will access Barista. Set their unmask to 0133.
3. Change the group for all Barista and Addon data to "barista".

Thin Client Without Web Start

You don't have to use Web Start to get a thin client two-tier configuration, it's just the easiest. If you want to, you download just the thin client BBj onto the client and use the following as a startup script or shortcut (each part that is on its own line should be separated with spaces. Parts in angle brackets <> should be changed to their correct values. The angle brackets are dropped.):

```
bbj
-RH<remote_host_name>
-tT0
-q
-WD<path/to/barista/on/remote/host>
-c</path/to/barista/config/file/>barista.cfg
</path/to/barista/programs/>bar_login.bbj
-
aon
-lENU;ESP
```

Three Tier, Data

To go to three tier you need to put the data files on another server. This computer will hold the Filesystem. In BBj there is no "data server" per se but you may hear people refer to "data server syntax". This is that familiar "<server_name,port=2000>" syntax in the Prefix or the data paths. Technically there does not need to be three separate servers for three tier (one can act as two of the tiers) but this is the most common setup.

- Put Addon data on a third server. This server needs to connect via TCP/IP to the server with BBjServices running (usually when the programs are). Use fixed IPs and hosts files as above.
- If this is a Linux box you will need to setup .rhosts files as above. Each login name from the client will need to have a group of "barista" (or whatever you want to name it) and permissions will need to be set to allow users with the group barista to access the data.
- Allow port 2000 through the firewalls between the AppServer (programs, BBjServices) and the Filesystem ("data server").
- On the server with the config files (the AppServer) you will need to set the configuration so it will look at the server with the filesystem (data server). There are lots of ways of doing this but the easiest is if all your application data is in one folder (say /usr/local/apps/aon/data) then put this in you Prefix in the barista.cfg file:

prefix /<server_name,port=2000>/usr/local/apps/aon/data/ ...

barista.cfg is usually found in <barista_home>/sys/config/enu

- In the barist.cfg file, look for all the STBLs that have aon/data in them. Make them point to nothing. For instance:

  SET +DIR_DAT=
  SET +ADDATA=
  SET +APDATA=
  etc.
- If you have separate data folders for your data files, don't put the data server syntax into the prefix; instead put it into all the STBLs.
  SET +DIR_DAT=/<server_name,port=2000>/usr/local/apps/aon/data/
  SET +ADDATA=/<server_name,port=2000>/usr/local/apps/aon/data/ad/
  SET +APDATA=/<server_name,port=2000>/usr/local/apps/aon/data/ap/
  etc.

- You will want to put all this into your *.syn files so that you will get your barista.cfg back after syncing. Your *.syn files can be anywhere but the default is apps/aon/config.