# Ask-a-Muse: How Do I Upgrade the JVM?

Posted At : June 11, 2009 2:41 PM | Posted By : Mark Kruger
Related Categories: ColdFusion, Coldfusion Troubleshooting

> *Muse Reader Rob Asks:*
> I have a silly question. How exactly *do* you upgrade the JVM on your ColdFusion server? My server is on Win2k3 x64 and the JVM version is 1.6.0_04. Do you specify it manually in the jvm.config file?

I'm glad you asked this question because it reminds me that I sometimes give advice without any follow through - which is the same problem I have with my 8 iron. Upgrading the JVM on a windows installation is pretty easy. Just remember that you will need the correct Java Runtime for your platform and ColdFusion version. Rob specified Win2k3 x64 so I assume he means he is running ColdFusion 8 enterprise 64 bit - in which case the target version is 1.6 update 14 (or 1.6.0_14). I usually start at the Sun Java **download** page. Once you have the right version in hand the rest is easy.

- Install it somewhere convenient. For my local dev workstation I put it in C:\jdk1.6.0_14\.
- Open the jvm.config file. If you don't know about this file shame on you. It means you have not adequately tuned your ColdFusion server.
- Find the line near the top that starts with ***java.home***=. I usually copy this line and then comment it out so I have an easy rollback plan. The value given after the equal sign will be the location of your current Java runtime. If you have CF installed in a typical configuration it will say *C:/ColdFusion8/runtime/jre*
- Change the value after the equal sign to point to the JRE folder of your new installation. For example, mine says *C:/jdk1.6.0_14/jre*.
- Restart the ColdFusion service.
- If it restarts successfully log into the cfide administrator and take a look at the "settings" page (the "I" icon at the top right). You should see the new Java version as 1.6.0_14.
- If the service restart fails take a peek in /coldfusion8/runtime/logs/*.out.log. It will probably have a clue in the last entry or two.

There are two more things to note. First, you may have noticed that the path specified in jvm.config uses front slashes instead of back slashes. This is one of the nuances of Java on Windows. Java treats the front slash as a path delimiter, but it requires the back slash to be escaped (at least in ini and properties files). So you *could* use backslashes, but if you did it would look like C:\\jdk1.6.0_14\\jre\\.

Second, you really want to have a rollback plan at the ready. Minor build upgrades are typically quite safe (as in moving from 1.6.0_4 to 1.6.0_14), but major upgrades and downgrades (as in moving from 1.5 to 1.6 or visa versa) have a tendency to produce unexpected results. If at all possible test the new version before rolling it on a production server. That's good advice for pretty much anything that functions at the system level.

### Additional Resource

Finally, my good friend and colleague and all around Linux smarty pants, Ryan Still has

posted these instructions on upgrading your JVM that include some great information on upgrading a Linux installation.