

Optimizing Tomcat for Web Applications

* TOMCAT *

Tomcat Security and Optimization Document

© PlanetJ Corporation
1821 Kelson Place, Escondido CA 92029
Phone 760-432-0600 • Fax 760-432-0600

Table of Contents

Introduction to Tomcat	3
Increase Maximum Memory Used	3
Changes to web.xml configuration file	5
Administering Tomcat:	8
Changing Port Settings	11
Secure Application Files in Tomcat	12
SSL:	15
Other Security Considerations	15
References	15

Introduction to Tomcat:

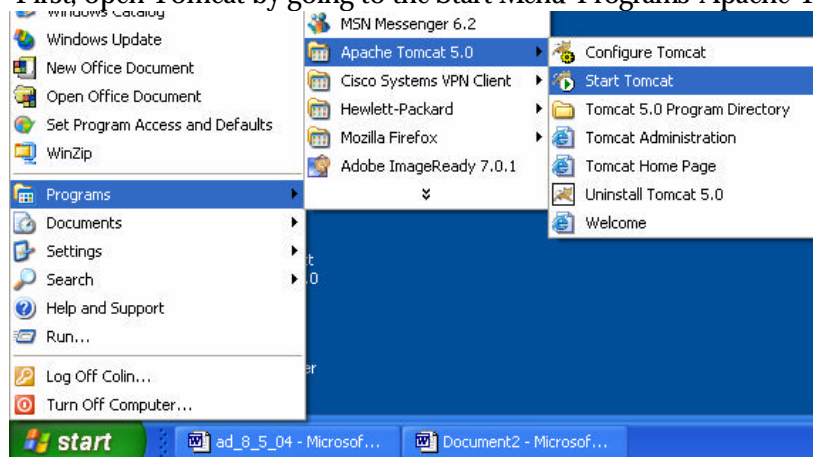
WebSphere has gained significant market share in Web application development space. A powerful, easy to use, and free alternative to WebSphere is Apache's Tomcat. Tomcat is open source and is the reference implementation for Sun' servlet standard. Tomcat has proven itself as production quality and is used by many companies. Tomcat installation on Window servers is an automated and easy process however the default installation places Tomcat into development mode. The advantage of development mode is that changes to JSPs, class, and JAR files are automatically incorporated at runtime without the need to restart the application server. The drawback is that performance suffers in development mode. Once you have created your WOW application and are ready to serve it to the public, the following steps should be taken to optimize Tomcat.

S T E P 1 ---

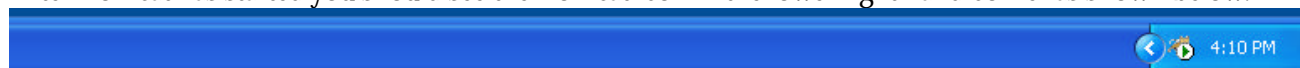
Increase Maximum Memory Used

Tomcat ships with the maximum amount of memory set to 64MB, which is not large enough to operate most large web applications. We recommend increasing the maximum amount of memory to 80% of the total RAM available on your server. This can be done as shown below.

First, open Tomcat by going to the Start Menu-Programs-Apache Tomcat-Start Tomcat.



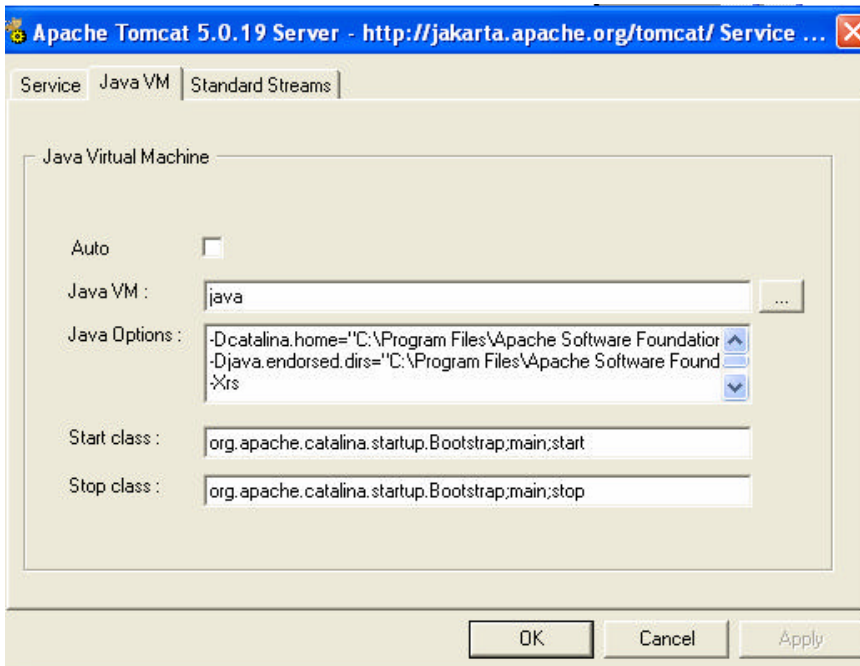
After Tomcat has started you should see the Tomcat icon in the lower right hand corner as shown below.



To access the Tomcat properties allowing changes to the maximum memory, right click on the icon and select properties. This is also the way to stop the server by clicking on Shutdown: Tomcat instead of properties.

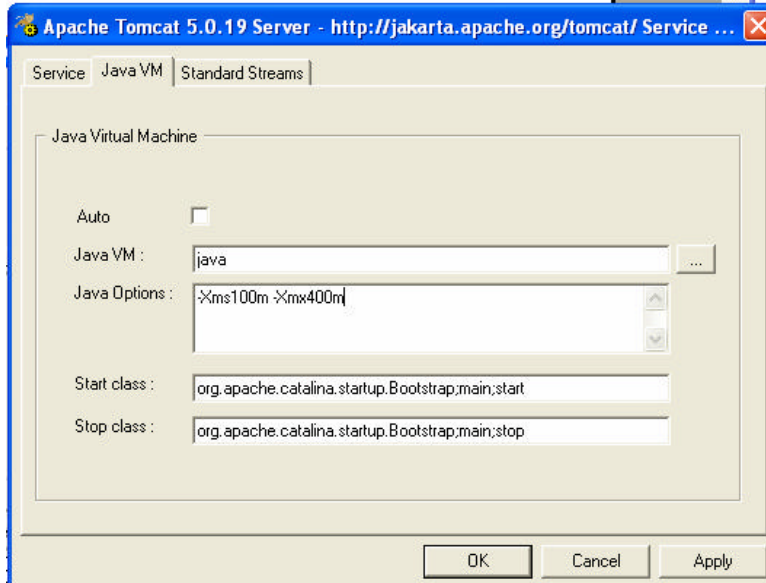


The properties screen below has the Java VM tab selected. Inside the Java Options we will add the increased min and max memory allocation settings.



The minimum value can be anything but should start with at least 64-100MB. The maximum memory should be up to 80% of your computers RAM. For example, if you have 512MB of RAM then you should set the maximum memory to around 400MB. To change the memory specifications you need to add this code to the Java Options text area:

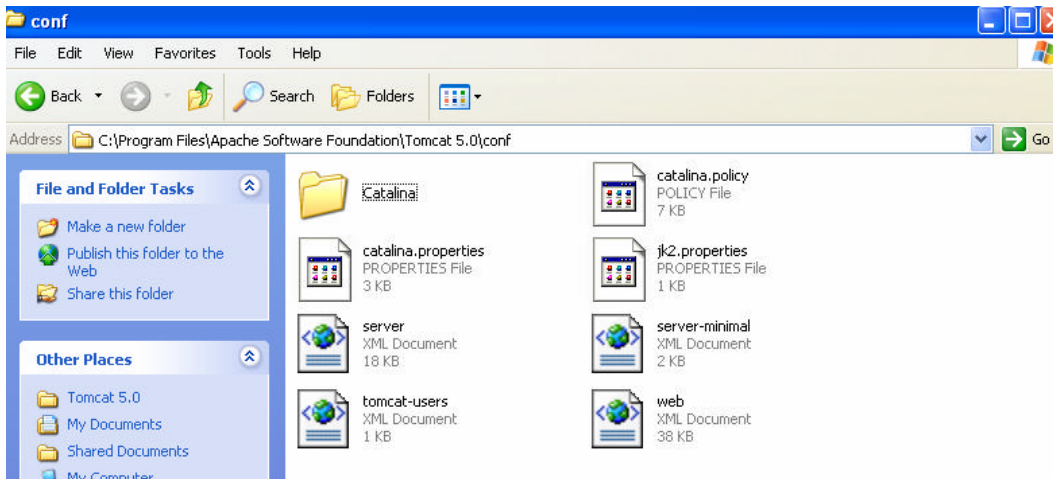
-Xms100m -Xmx400m



STEP 2

Changes to web.xml configuration file:

The web.xml is the document that defines default values for **all** web applications loaded into each instance of Tomcat. As each application is deployed, this file is processed, followed by the "/WEB-INF/web.xml" deployment descriptor from your own applications. The web.xml is located in Apache Software Foundation-Tomcat-conf folder shown below.



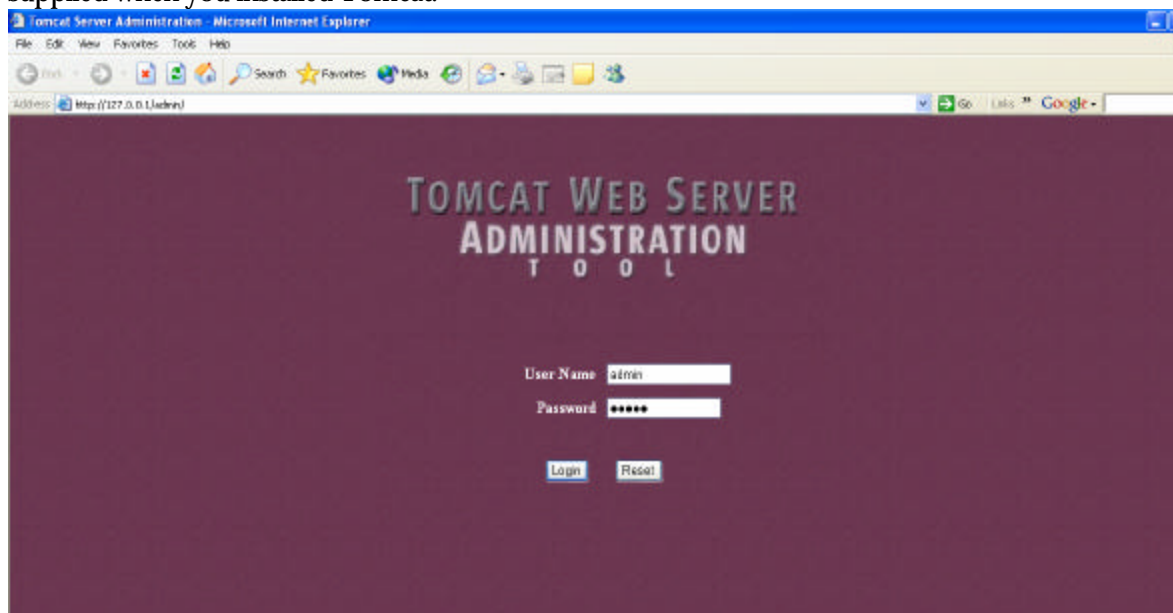
In summary, Tomcat is a cost-efficient tool to hosting Web applications. Now that you have made the above simple changes, Tomcat will run much faster and handle more complex workloads.

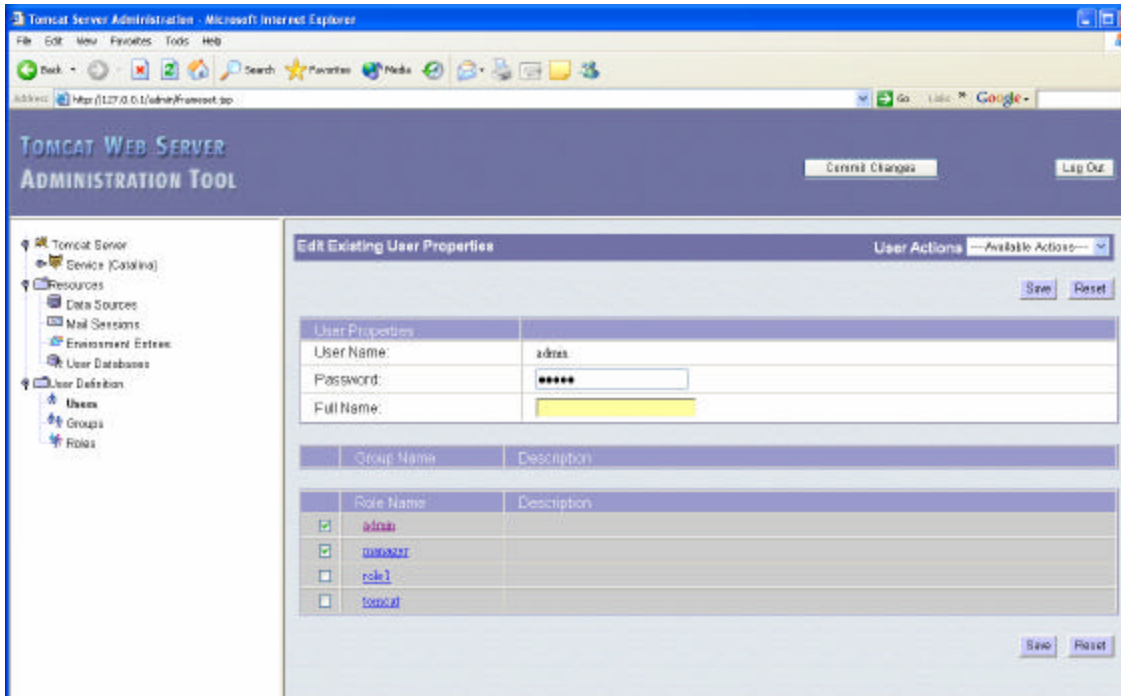
Administering Tomcat:

Tomcat also has its own administration, which can be helpful and accessed through the Tomcat server. Now we are going to show some of the helpful and powerful administration applications that Tomcat ships with such as Tomcat Administrator and Tomcat Manager.

To access both of these you can either go through the start menu under Apache Tomcat folder or you can open a browser and type go to the localhost such as <http://localhost:80> with the port that you specified in the installation of Tomcat.

Tomcat allows administration (start, stop, destroy, manage, create etc) via a web application. The Tomcat Administration Application allows you to see the server information, edit connections, edit users, groups, resources and change the main settings of the webserver. This application runs using the userid and password supplied when you installed Tomcat.





The userid and pwd are defined in tomcat at:

..../config/tomcat_users.xml as shown below.

```
<?xml version="1.0" encoding="utf-8" ?>
<tomcat-users>
<role rolename="tomcat" />
<role rolename="role1" />
<role rolename="manager" />
<role rolename="admin" />
<user username="tomcat" password="tomcat" roles="tomcat" />
<user username="role1" password="tomcat" roles="role1" />
<user username="both" password="tomcat" roles="tomcat,role1" />
<user username="admin" password="admin" roles="admin,manager" />
</tomcat-user>
```

These are set when you first install Tomcat on to your machine when the installation asks for username and password. You can also change the username and password and add new users in Tomcat Administration Application.

Tomcat also ships with the Tomcat Manager. This is an application that will allow you to edit, start, and stop applications dynamically without restarting the Tomcat. It also shows current applications and a few other options, which you can learn about by clicking on the Manager Help Link Shown below.

Manager - Microsoft Internet Explorer

Address: http://localhost:8080/manager/html

The Apache Jakarta Project
http://jakarta.apache.org/

Tomcat Web Application Manager

Message: OK

Manager

List Applications HTML Manager Help Manager Help Server Status

Path	Display Name	Running	Sessions	Commands			
/	Welcome to Tomcat	true	0	Start	Stop	Reload	Undeploy
/admin	Tomcat Administration Application	true	1	Start	Stop	Reload	Undeploy
/balancer		true	0	Start	Stop	Reload	Undeploy
/jsp-examples	JSP 2.0 Examples	true	0	Start	Stop	Reload	Undeploy
/manager	Tomcat Manager Application	true	0	Start	Stop	Reload	Undeploy
/servlets-examples	Servlet 2.4 Examples	true	0	Start	Stop	Reload	Undeploy
/tomcat-docs	Tomcat Documentation	true	0	Start	Stop	Reload	Undeploy
/wiki	WikiSite Content Management	true	0	Start	Stop	Reload	Undeploy
/wsnRO	WOW	true	0	Start	Stop	Reload	Undeploy

Deploy

Deploy directory or WAR file located on server:

Context Path (optional):

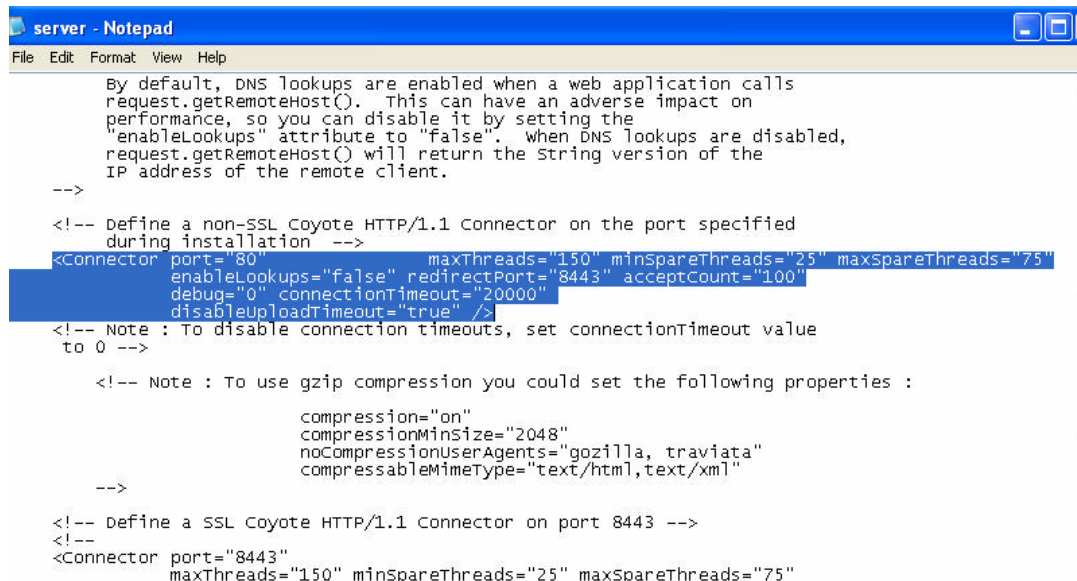
XML Configuration file URL:

WAR or Directory URL:

These are a few more tools to help you understand and optimize Tomcat for your Web Applications and to use WOW.

Changing Port Settings:

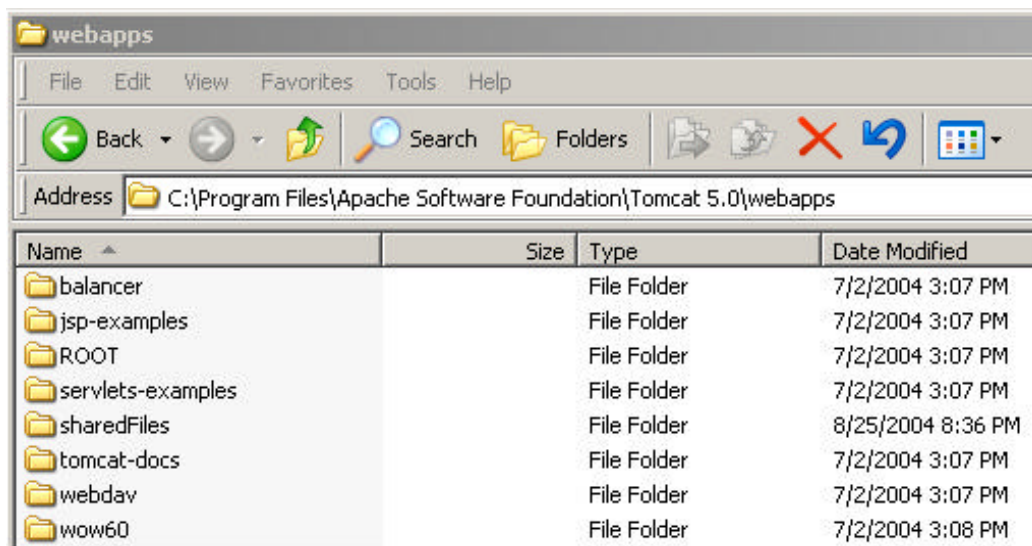
When you install Apache Tomcat it defaults the port to 8080. The ports are set in the `..Tomcat/conf/server.xml` document. To adjust the port being used right click and open with notepad or any other text editor. After the document is open scroll down around half way looking for the connector tag shown below and change the port number to a new port number (Usually either 8080 or 80). Then save the newly changed `server.xml` file and restart Tomcat.



```
server - Notepad
File Edit Format View Help
By default, DNS lookups are enabled when a web application calls
request.getRemoteHost(). This can have an adverse impact on
performance, so you can disable it by setting the
"enableLookups" attribute to "false". When DNS lookups are disabled,
request.getRemoteHost() will return the String version of the
IP address of the remote client.
-->
<!-- Define a non-SSL Coyote HTTP/1.1 Connector on the port specified
during installation -->
<Connector port="80" maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
enableLookups="false" redirectPort="8443" acceptCount="100"
debug="0" connectionTimeout="20000"
disableUploadTimeout="true" />
<!-- Note : To disable connection timeouts, set connectionTimeout value
to 0 -->
<!-- Note : To use gzip compression you could set the following properties :
compression="on"
compressionMinSize="2048"
noCompressionUserAgents="gozilla, traviata"
compressableMimeType="text/html,text/xml"
-->
<!-- Define a SSL Coyote HTTP/1.1 Connector on port 8443 -->
<!--
<Connector port="8443"
maxThreads="150" minSpareThreads="25" maxSpareThreads="75"
```

Secure Application Files in Tomcat:

When installing a new web application such as WOW to Tomcat, Tomcat automatically allows people to view the directory listing of all files inside of the webapps folder. An example of this is the **sharedFiles** folder in the webapps folder on the hard drive shown below.



By Default the **sharedFiles** folder includes three directories: one directory contains sample images and another directory contains sample MS Word Documents. These folders are for example purpose only and may be deleted or added upon. There is also a directory labeled **WEB-INF**. This folder contains a file called web.xml. Web.xml allows WOW to be set up to show its contents over the Internet or to be set so that those files cannot be seen over the Internet. Below is an example of connecting to the **sharedFiles** directory through Tomcat.

Directory Listing For /

Filename	Size	Last Modified
HollyCow2.html	0.1 kb	Fri, 09 Apr 2004 18:46:34 GMT
QueryBuilder.hta	0.3 kb	Wed, 21 Apr 2004 22:08:18 GMT
QueryBuilder.html	0.3 kb	Wed, 21 Apr 2004 21:56:42 GMT
Thumbs.db	10.0 kb	Tue, 24 Aug 2004 00:00:52 GMT
images/		Fri, 20 Aug 2004 22:06:26 GMT
jc/		Sun, 22 Aug 2004 01:31:08 GMT
tomcat-power.gif	2.2 kb	Sat, 14 Feb 2004 10:21:44 GMT
tomcat.gif	1.8 kb	Sat, 14 Feb 2004 10:21:44 GMT

Apache Tomcat/4.1.24

This creates a security problem, as the whole Internet will be able to see our code and mess around with our files. To disable this feature you need to go in Apache Tomcat file structure <Apache Tomcat>/conf/web.xml file. In this web.xml file shown below there is an initial parameter called listing that is automatically set to true. We want to change that to false so that your files cannot be directly accessed and seen over the Internet.

SSL:

To set up SSL and other security features for Tomcat refer to the Apache's Tomcat site for SSL setup and optimization, <http://jakarta.apache.org/tomcat>

Other Security Considerations:

1. Make sure you have set proper userids and passwords for your Tomcat user file. This file is located at:`/config/tomcat_users.xml`

References

Tomcat: <http://jakarta.apache.org/tomcat>

PlanetJ's WOW: <http://www.planetjavainc.com>