## 部署tomcat方式

1. **项目直接放入 webapps 目录中**

**2，修改 conf/server.xml 文件**

打开tomcat下conf/server.xml，在<Host> </Host>标签之间输入项目配置信息：<Context path="/WebProject" docBase="D:/WebProject" reloadable="true" />

　　path:浏览器访问时的路径名

　　docBase:web项目的WebRoot所在的路径，注意是WebRoot的路径，不是项目的路径。其实也就是编译后的项目

　　reloadble:设定项目有改动时，tomcat是否重新加载该项目

**3，apache-tomcat-7.0.52\conf\Catalina\localhost**

进入到 apache-tomcat-7.0.52\conf\Catalina\localhost 目录，新建一个 项目名.xml 文件

在 那个新建的 xml 文件中，增加下面配置语句（和上面的是一样的,但是不需要 path 配置，加上也没什么用）

<Context  docBase="D:/WebProject" reloadable="true" />

### 总结：

①、第一种方法比较普通，但是我们需要将编译好的项目重新 copy 到 webapps 目录下，多出了两步操作

②、第二种方法直接在 server.xml 文件中配置，但是从 tomcat5.0版本开始后，server.xml 文件作为 tomcat 启动的主要配置文件，一旦 tomcat 启动后，便不会再读取这个文件，因此无法再 tomcat 服务启动后发布 web 项目

③、第三种方法是最好的，每个项目分开配置，tomcat 将以\conf\Catalina\localhost 目录下的 xml 文件的文件名作为 web 应用的上下文路径，而不再理会 <Context>中配置的 path 路径，因此在配置的时候，可以不写 path。

通常我们使用第三种方法

## 公司部署项目

1. 停止tomcat服务
2. 导入war包到webapp下，备份原项目文件夹下data，MG，projects；删除原项目文件夹。

3，解压war包，导入data，MG，projects；重启tomcat

## Window下设置tomcat为服务

Window下：管理员运行cmd，cd到tomcat的bin目录下，执行service install

## **Linux下配置多个tomcat**

1. 修改catalina.sh文件，设置本tomcat使用的环境变量；

#在第一大块注释下写，JAVA\_HOME和这一个Tomcat自身的目录 JAVA\_HOME=/usr/local/java/jdk/jdk1.7.0\_80

CATALINA\_HOME=/usr/local/tomcat/tomcat-8090

1. 设置服务并自启

复制catalina.sh到系统启动目录下并个性化命名

[root@localhost bin]# cp /usr/local/tomcat/tomcat-8090/bin/catalina.sh /etc/rc.d/init.d/tomcat-8090

1. 修改tomcat-8090文件的启动/关闭级别

#!/bin/sh

# chkconfig: 35 70 70

# description: tomcat-8090 service desc

他们的意义我在另一篇博文中有介绍过，想看的http://blog.csdn.net/u010297957/article/details/50751656

1. 将tomcat-8090服务配置自启，并加入linux服务监控

[root@localhost bin]# chkconfig --add tomcat-8090

此时，就可以在rc.d下的相应的3和5目录下看到S70tomcat-8090，并在0、1、2、4、6目录看到K70tomcat-8090了

已经完成了，下面是一些手动的命令

1. reboot linux系统后，你就会看到tomcat-8090自己已经启动了

2. service tomcat-8090 start/shutdown/restart

## Tomcat环境配置Linux、Window

Windows：例如tomcat安装在 E:\tomcat；需要在环境变量配置：

CATALINA\_HOME=E:\tomcat

CATALINA\_BASE= E:\tomcat

Linux：例如tomcat安装在：/usr/local/tomcat8.5

 [root@ora-rac2 ~]# vi /etc/profile  进入配置文件，添加下列环境变量，编辑完Esc -> :wq 保存退出

 export JAVA\_HOME=/cdr/jdk1.7.0\_51

export PATH=$JAVA\_HOME/bin:$PATH

 Export CLASSPATH=.:$JAVA\_HOME/lib/tools.jar:$JAVA\_HOME/lib/dt.jar:$CLASSPATH    前三行是java环境配置

**export CATALINA\_HOME= /usr/local/tomcat8.5**

**export CATALINA\_BASE= /usr/local/tomcat8.5**   后面这两行是tomcat环境配置

保存后，退出

**[root@ora-rac2 ~]# source /etc/profile**            牢记，这部一定要执行！！！！！

关闭当前连接的会话，即可生效。

tomcat如果嵌入到开发环境如eclipse直接在开发环境中直接运行即可，无需配置环境；

           如果单独从命令行启动tomcat，必须配置上述环境！             **（建议都配置环境防止运行出错）**

## **修改tomcat的JVM参数**

Window下：运行tomcat的bin目录下tomcatXw.exe，Java条目下设置。

Linux下：

要添加在tomcat 的bin 下catalina.sh 里，位置cygwin=false前 。注意引号要带上,红色的为新添加的.

# OS specific support. $var \_must\_ be set to either true or false.  
JAVA\_OPTS="-Xms256m -Xmx512m -Xss1024K -XX:PermSize=128m -XX:MaxPermSize=256m"  
cygwin=false

在第一行加入

JAVA\_OPTS="-Xms36m -Xmx36m -Xss256K -XX:ParallelGCThreads=8 -XX:+UseConcMarkSweepGC -Xloggc:/var/tomcat\_gc.log"

JAVA\_OPTS   所有的java都受影响

CATALINA\_OPTS="-Xms36m -Xmx36m -Xss256K -XX:ParallelGCThreads=8 -XX:+UseConcMarkSweepGC -Xloggc:/var/tomcat\_gc.log"

CATALINA\_OPTS 只有此tomcat接受参数影响