

tomcat 学习 | tomcat 的启动过程

作者: [xiaodaojava](#)

原文链接: <https://ld246.com/article/1565188617172>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<h2 id="开头说两句">开头说两句</h2>

<p>小刀博客: https://www.lixiang.red

小刀公众号: 程序员学习大本营</p>

<h2 id="学习背景">学习背景</h2>

<p>在前两篇中,我们讲述了 tomcat 的源码结构

https://www.lixiang.red/articles/2019/08/04/1564907616303.html

还有 tomcat 的 server.xml 结构.

https://www.lixiang.red/articles/2019/08/06/1565094623138.html

今天我们来继续看 tomcat 的启动过程</p>

<h2 id="从已知startup-bat-sh入手">从已知 startup.bat/sh 入手</h2>

<p>windows 上启动是:startup.bat

linux/mac 上启动是 startup.sh

 </p>

<h2 id="startup-sh">startup.sh</h2>

<p>重点在于最后一行:</p>

```
<pre> <code class="highlight-chroma"> <span class="highlight-line"> <span class="highlight-cl"># PRGDIR 是当前tomcat下的bin目录
```

```
</span> </span> <span class="highlight-line"> <span class="highlight-cl">PRGDIR=`dirname "$PRG`
```

```
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
```

```
</span> </span> <span class="highlight-line"> <span class="highlight-cl">EXECUTABLE=catalina.sh
```

```
</span> </span> <span class="highlight-line"> <span class="highlight-cl">
```

```
</span> </span> <span class="highlight-line"> <span class="highlight-cl">执行tomcat/bin/catalina.sh start
```

```
</span> </span> <span class="highlight-line"> <span class="highlight-cl">@$@ 是代表全部的数,
```

```
</span> </span> <span class="highlight-line"> <span class="highlight-cl">exec "$PRGDIR"/EXECUTABLE" start "$@"
```

```
</span> </span> </code> </pre>
```

<h2 id="catalina-sh">catalina.sh</h2>

<p>我们通过参数 start 进行追踪:

在这里看到了对 <code>start</code> 参数进行判断,然后走不同的分支逻辑

经过一大堆的判断,最后达到可启动的状态时,就开始执行启动的命令:

 </p>

<p>nohup: 是 linux 系统中,可以后台运行程序的命令,窗口关掉也会存在</p>

<p>ENDORSED_PROP: 可以覆盖部分 jvm bootstarp 类加载器加载的类</p>

<p>org.apache.catalina.startup.Bootstrap: 最后会执行这个类</p>

<h2 id="Bootstrap">Bootstrap</h2>

<p>根据上文,我们可以找到 Bootstrap 这个类的主函数

如上图所示,只要执行了以下几步:</p>

初始化 bootstrap 实例

- 调用 bootstrap 的 init 方法,去初始化类加载器,以及 catalina 实例
- 调用 bootstrap 的 start 方法,然后通过反射去调用 catalina 的 start 方法

<p>init():</p>

```

public void init() throws Exception {
    // 初始化类
    // 设置当前
    // 当前线程:
    // Load our s
    // 找到到 Cata
    // Set the sha
    // 获取到Catal
    // 通过反射
}

```

置 Catalina实例使用的父类加载器为sharedLoader

```
</span></span><span class="highlight-line"><span class="highlight-cl"> Method met  
od =  
</span></span><span class="highlight-line"><span class="highlight-cl"> startupInst  
nce.getClass().getMethod(methodName, paramTypes);  
</span></span><span class="highlight-line"><span class="highlight-cl"> method.invo  
e(startupInstance, paramValues);  
</span></span><span class="highlight-line"><span class="highlight-cl"> catalinaDae  
</span></span><span class="highlight-line"><span class="highlight-cl"> on = startupInstan  
ce;  
</span></span><span class="highlight-line"><span class="highlight-cl">  
</span></span><span class="highlight-line"><span class="highlight-cl"> }  
</span></span></code></pre>
```

<p>start():</p>

```
<pre><code class="highlight-chroma"><span class="highlight-line"><span class="highlight" style="color: red;">public void start()  
</span></span><span class="highlight-line"><span class="highlight-cl"> throws Excep  
tion {  
</span></span><span class="highlight-line"><span class="highlight-cl"> if( catalinaDa  
emon==null ) init();  
</span></span><span class="highlight-line"><span class="highlight-cl"> // 使用反射  
调用Catalina 的 start 方法  
</span></span><span class="highlight-line"><span class="highlight-cl"> Method met  
od = catalinaDaemon.getClass().getMethod("start", (Class [] )null);  
</span></span><span class="highlight-line"><span class="highlight-cl"> method.invo  
e(catalinaDaemon, (Object [])null);  
</span></span><span class="highlight-line"><span class="highlight-cl">  
</span></span><span class="highlight-line"><span class="highlight-cl"> }  
</span></span></code></pre>
```

<h2 id="最后说两句">最后说两句</h2>

<p>这样就完成了 tomcat 从命令行到 bootstrap 以及 catalina 的初步初始化.后面还有 server,service, Engine, host 等组件的加载

在本篇学习中,小伙伴们有什么问题,可以和小刀一起交流: best396975802</p>