



嵌入式Tomcat

T H A N K Y O U F O R W A T C H I N G

主讲老师：King QQ: 2962938812



目录

CONTENTS



嵌入式Tomcat

介绍
优点
发展趋势



Maven集成 Tomcat插件

配置
运行
分析



Tomcat API接口

API接口



手写嵌入式Tomcat

手写思路分析
手写嵌入式Tomcat



嵌入式Tomcat:

非传统的部署方式，将Tomcat嵌入到主程序中进行运行。

优点:

灵活部署、任意指定位置、通过复杂的条件判断。

发展趋势:

Springboot默认集成的是Tomcat容器。

Maven中Springboot引入Tomcat

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-tomcat</artifactId>
    <scope>provided</scope>
</dependency>
```

02 Maven集成Tomcat插件



Tomcat 7 Maven插件: [tomcat7-maven-plugin](#)

```
<dependency>
  <groupId>org.apache.tomcat.maven</groupId>
  <artifactId>tomcat7-maven-plugin</artifactId>
  <version>2.2</version>
</dependency>
```

插件运行

选择pom.xml文件，击右键——>选择 Run As——> Maven build
在Goals框加加入以下命令: tomcat7:run

Maven集成Tomcat 插件启动Tomcat是常用的Tomcat嵌入式启动

03 Maven集成Tomcat插件启动分析



分析它的启动

Tomcat7RunnerCli是引导类

进一步分析

Tomcat7RunnerCli主要依靠Tomcat7Runner

分析结论

原来嵌入式启动就是调用了Tomcat的API来实现的



实现嵌入式Tomcat的基础:

Tomcat本身提供了外部可以调用的API

Tomcat类:

1. 位置: `org.apache.catalina.startup.Tomcat`
2. 该类是public的。
3. 该类有Server、Service、Engine、Connector、Host等属性。
4. 该类有`init()`、`start()`、`stop()`、`destroy()`等方法。

Tomcat
+ Server + Service + Engine + Connector + Host
+ init() + start() + stop() + destroy()

分析结论:

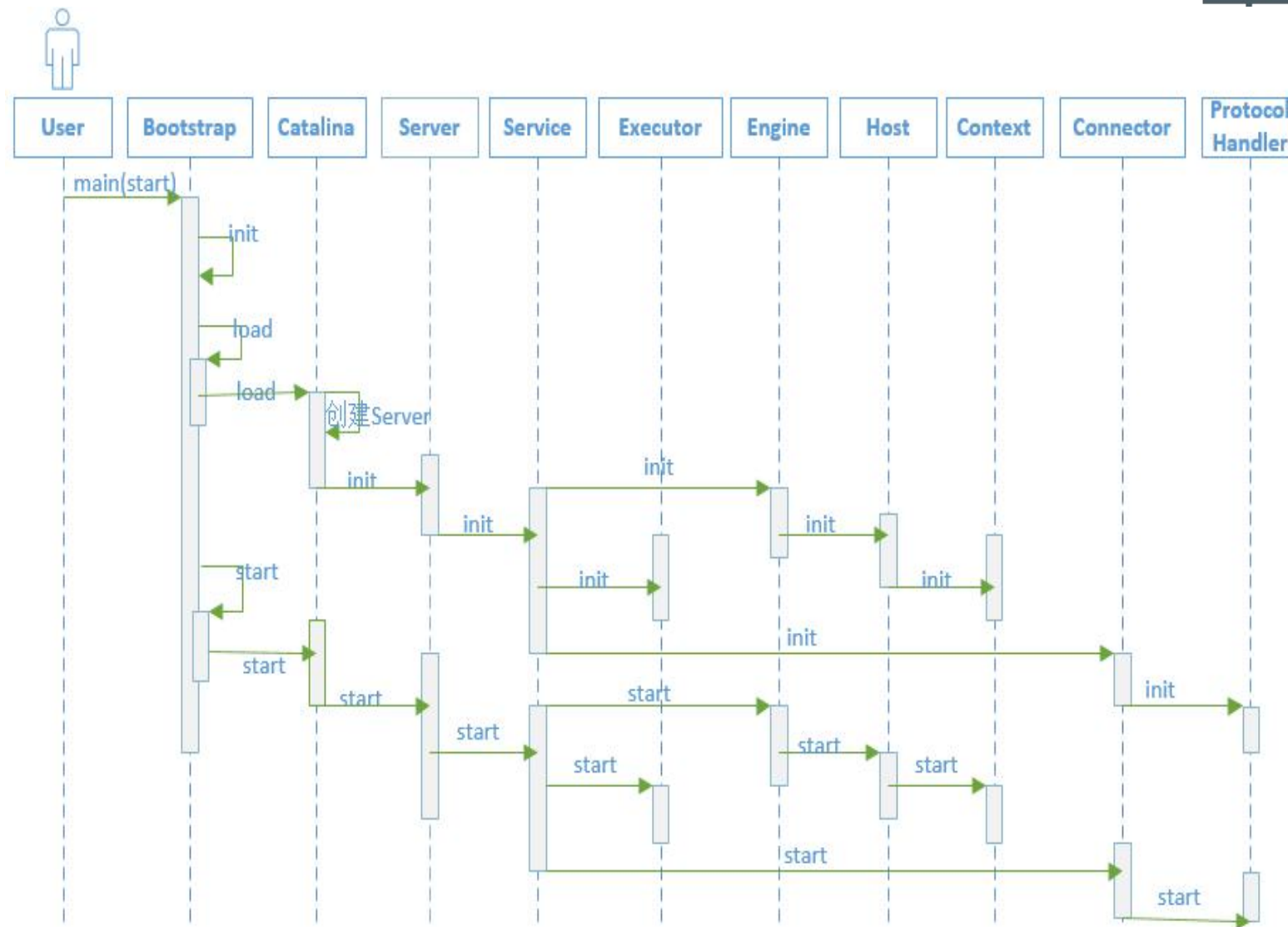
Tomcat类是外部调用的入口



分析：
Tomcat单独启动时的流程

结论：
使用Tomcat的API来实现：

1. 新建一个Tomcat对象
2. 设置Tomcat的端口号
3. 设置Context目录
4. 添加Servlet容器
5. 调用Tomcat对象Start ()
6. 强制Tomcat等待





1. 准备好一个简单的Servlet项目
2. 新建一个手写嵌入式Tomcat工程
3. Tomcat工程中使用一个类完成手写嵌入式Tomcat的功能
4. 调用该类的main方法执行
5. 效果演示和分析



完 毕！

享学课堂：King老师