



# Tomcat性能优化

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

---



### 性能优化

---

指标  
原则  
工具



### Tomcat优化

---

server.xml优化  
web.xml优化



### SpringBoot 中Tomcat优化

---

设置线程池  
关闭Accesslog



性能优化已经成为架构师的标配！

性能优化的三个指标

降低响应时间

提高系统吞吐量(QPS)

提高服务的可用性

性能优化的原则

具体情况具体分析

积少成多



**JConsole:** 一个内置 Java性能分析器

jdk安装目录/bin/jconsole.exe

**JMeter:** 一个Apache组织开发的基于Java的压力测试工具

下载地址:[http://jmeter.apache.org/download\\_jmeter.cgi](http://jmeter.apache.org/download_jmeter.cgi)

## 性能优化测试原则

指标: 正确率、CPU占有率、QPS、JVM

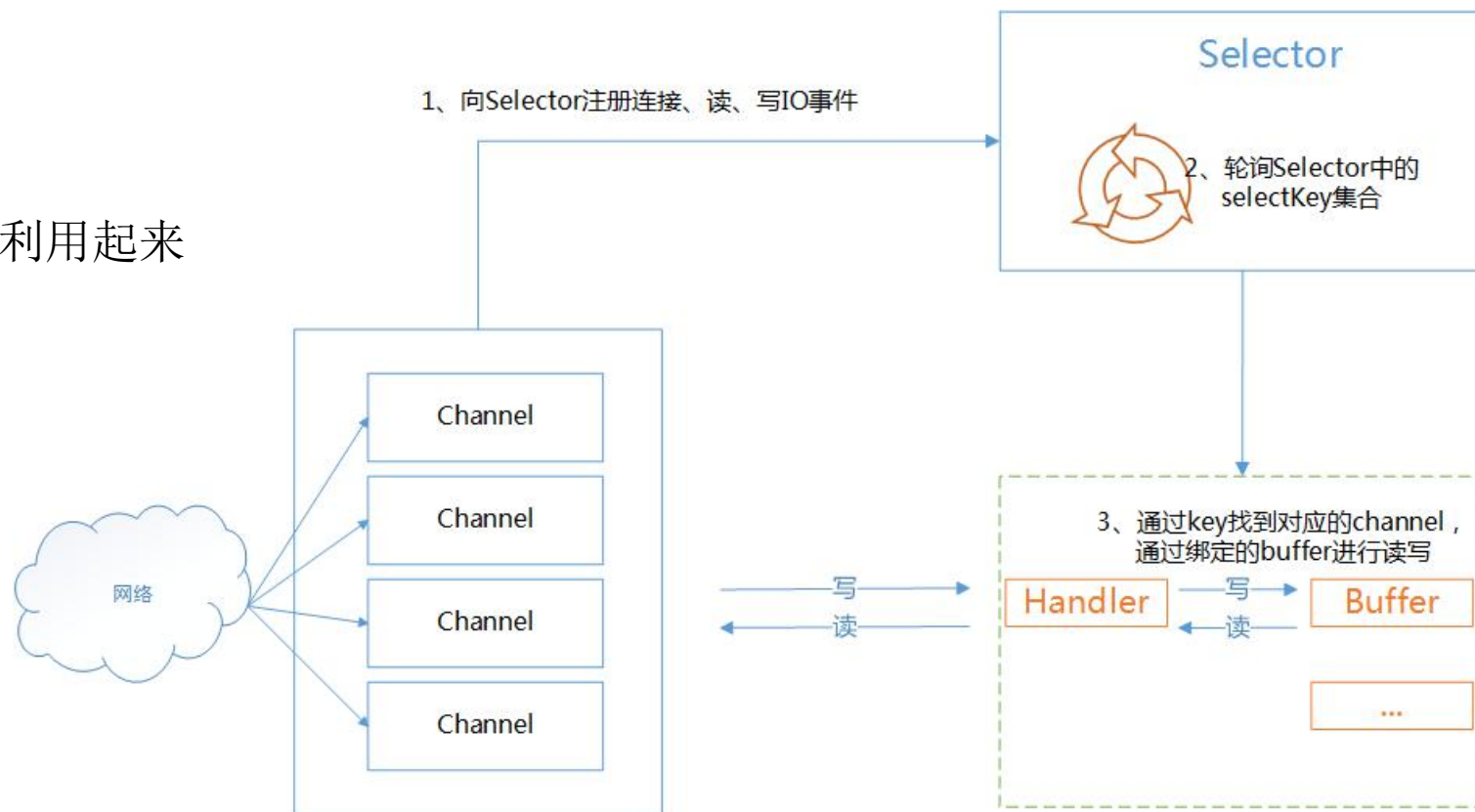
程序分类: I/O密集型(网络, 文件读写), CPU密集型(加密/解密算法)

## Connector连接器的I/O模型

连接器模式改为NIO模式

NIO模式最大化压榨了CPU，把时间片更好利用起来

NIO适合大量长连接





应用程序运行时最终会加载conf/web.xml和应用的web.xml的合集

## servlet优化

当前应用是REST应用(微服务):

1. 去掉不必要的资源: JspServlet
2. session也可以移除

## valve优化:

移除掉AccessLogValve

valve实现都需要消耗java应用的计算时间, 一般我们可以使用nginx来记录日志



## 关闭自动重载

```
<Context docBase="" reloadable="false"/>
```

关闭自动重载，默认是true(不同版本中有差异)

自动加载增加运行开销并且很容易内存溢出

## 配置线程池

Executor标签中属性

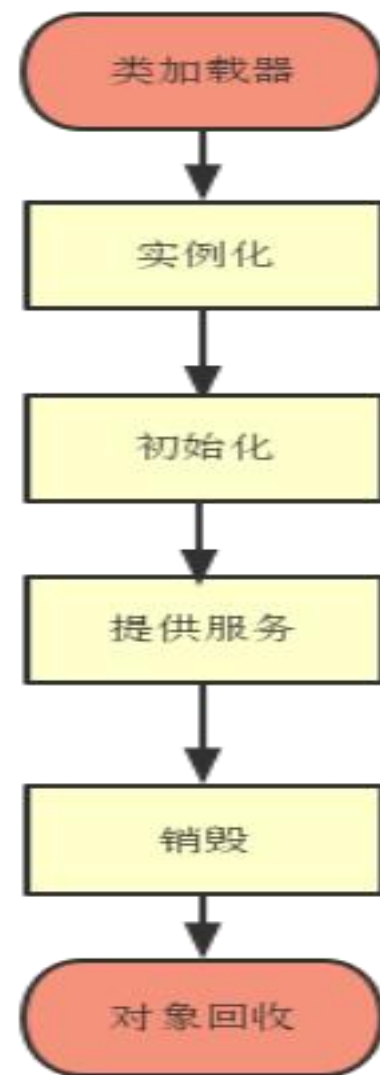
【namePrefix】线程命名前缀

【maxThreads】最大允许线程数

【minSpareThreads】最少空闲线程，相当于初始化的线程，线程池中的线程

Connector标签中的属性

【executor】对上面Executor标签标签的引用





## JSP预编译优化

JSP -> JAVA -> CLASS

可使用ant先编译jsp

JspServlet开发模式(development)设置为false





### Maven中Springboot引入Tomcat

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

#### 1. 设置线程池

```
server.tomcat.max-threads=1000
```

#### 2. 关闭Accesslog日志

```
server.tomcat.accesslog=false
```



# 完 毕！

享学课堂：King老师