

一、Tomcat处理请求过程

Tomcat通过Endpoint组件接收socket连接，接收到一个socket连接后会执行如下步骤

1. 第一次从socket中获取数据到InputBuffer中，BIO对应的是InternalInputBuffer，父类是AbstractInputBuffer
2. 然后基于InputBuffer进行解析数据
3. 先解析请求行，把请求方法，请求uri，请求协议等封装到org.apache.coyote.Request对象中
4. org.apache.coyote.Request中的属性都是MessageBytes类型，直接可以理解为字节类型，因为从socket中获取的数据都是字节，在解析过程中不用直接把字节转成字符串，并且MessageBytes虽然表示字节，但是它并不会真正的存储字节，还是使用ByteChunk基于InputBuffer中的字节数组来进行标记，标记字节数组中的哪个范围表示请求方法，哪个范围表示请求uri等等。
5. 然后解析头，和解析请求行类似
6. 解析完请求头后，就基于请求头来初始化一些参数，比如Connection是keepalive是close，比如是否有Content-length，并且对于的长度是多少等等，还包括当前请求在处理请求体时应该使用哪个InputFilter。
7. 然后将请求交给容器
8. 容器再将请求交给具体的servlet进行处理
9. servlet在处理请求的过程中会利用response进行响应，返回数据给客户端，一个普通的响应过程会把数据先写入一个缓冲区，当调用flush，或者close方法时会把缓冲区中的内容发送给socket，下面有一篇单独的文章讲解tomcat响应请求过程
10. servlet处理完请求后，先会检查是否需要把响应数据发送给socket
11. 接着看当前请求的请求体是否处理结束，是否还有剩余数据，如果有剩余数据需要把这些数据处理掉，以便能够获取到下一个请求的数据
12. 然后回到第一步开始处理下一个请求