链滴

用 Netty 造一个简单的 RPC

作者：614756773

原文链接：https://ld246.com/article/1576570517791

来源网站：链滴

<img src="https://ld246.com/images/img-loading.svg" alt="" data-src="https://b3logfile
com/bing/20190224.jpg?imageView2/1/w/960/h/540/interlace/1/q/100">
<h3 id="一-项目结构">一、项目结构</h3>
<h4 id="hotrpc-common">hotrpc-common</h4>

主要代码

自定义注解，用于标志代理类
netty 的相关操作，用于服务之间的通信

<code>netty client</code> 和 <code>netty server</code>
编码器，解码器

服务接口的定义以及对服务实现类的扫描注册

将服务实现类进行实例化
当客户端调用服务时，找到相应的实现类，用实现类进行代理

该子项目主要用于客户端和服务端之间的通信，自定义了一个请求协议，请求格式如下：
<table>
<thead>
<tr>
<th>header</th>
<th>dataLength</th>
<th>data</th>
</tr>
</thead>
<tbody>
<tr>
<td>614756773</td>
<td>variable</td>
<td>variable</td>
</tr>
</tbody>
</table>

一次完整的请求一共有 3 个部分

<code>header</code> 占 4 字节为固定值 <code>614756773</code>，作为一个 <code
魔数</code> 使用，以此来校验请求是否合法
<code>dateLength </code> 占 4 字节，为一个变量，用来表示 data 有多少字节
<code>data</code>，传输的数据

<h4 id="hotrpc-server">hotrpc-server</h4>

主要代码

定义服务实现类

<h4 id="hotrpc-client">hotrpc-client</h4>

主要代码

扫描标有 RpcCaller 注解的接口，并代理，然后注册成 bean 放入 spring 容器中
定义 controller，并且从 spring 容器中获取到 <code>代理服务</code>

<hr>
<h3 id="二-主流程">二、主流程</h3>
<h4 id="客户端">客户端</h4>

启动 SpringBootApplication
扫描标有 RPC 注解的类
注册到 spring 容器中

注册时使用 jdk 动态代理该类 <code>代理类东西有点多看下面</code>

从 spring 容器中获得该类，调用方法

jdk 动态代理过程

先将服务接口封装成一个 <code>Request</code> 对象，包括设置 <code>className</cod
>，<code>methodName</code>，<code>params</code> 等
实例化一个 <code>Client</code> 对象，然后通过该对象将请求发送出去
阻塞等待响应
从响应结果中获取数据并返回

<h4 id="服务端">服务端</h4>

实例化 <code>Server</code> 并且启动
检查 <code>Server</code> 状态，保证不会重复启动
把 rpc 服务实现类扫描出来然后实例化，缓存在 Map 对象中
若有请求来临

对请求进行解码
获取到请求体，从而找到对应的 rpc 服务实现类
调用服务实现类
对调用结果进行编码
发送结果给客户端

<hr>
<h3 id="三-优化点">三、优化点</h3>

目前如果需要定义一个 rpc 服务，必须在 <code>hotrpc-common</code> 里先定义接口，后再 <code>hotrpc-server</code> 中定义实现类。
可以改成 cglib 动态代理，这样就不必再定义接口了，也能让 <code>hotrpc-common</code> 装的更好
<code>hotrpc-client</code> 中需要定义如何去扫描 RpcCaller，如何进行代理。
这样做很不好，应该放在 <code>hotrpc-common</code> 包中去做这些事情。
由于进行扫描类操作，是使用的 spring 的工具类 <code>ClassPathScanningCandidateComponentProvider</code> 所以都写死了包路径为 <code>cn.hotpot</code>。
可以使用 <code>org.reflections</code> 工具类来进行扫描。

<hr>
<h4 id="源码">源码</h4>
github