ROP框架

①

SecurityManager

RopServlet

②

③

⑨

ServiceRouter

④

⑧

⑦

⑥

XXXService

ServiceMethodAdapter

RopMarshaller

⑤

1. ：客户端向服务端发送请求，当请求满足Rop配置的url匹配模式时，会进入Rop框架进行处理。

通过在web.xml中配置Rop框架的url匹配模式，如：

<servlet>

<servlet-name>rop</servlet-name>

<servlet-class>com.rop.RopServlet</servlet-class>

<load-on-startup>2</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>rop</servlet-name>

<url-pattern>/router</url-pattern>

</servlet-mapping>

Remark：Rop框架是基于Spring框架的，需要配置Spring容器配置文件，并在该配置文件中增加Rop命名空间定义。

1. ：RopServlet接收到请求后，直接调用ServiceRouter的默认实现：

AnnotationServletServiceRouter的service()方法。：

1. ：ServiceRouter会调用SecurityManager进行安全性验证，验证的顺序为：先检查系统参数的正确性，依次是：appKey(是否有appKey参数,若有，参数值是否有效) -> 会话检查(判断调用方法是否需要在会话状态下调用，若是，检查是否有sessionId,若有sessionId，检查对应的值是否有效) -> 检查method参数(检查有没有method参数，若有，则对应的值是否在集合Set<String>中) -> 检查v参数(检查有没有，有就与method联合起来判断是否在字典中，key为method+"#"+v) -> 检查签名正确性(是否开启了签名验证，若开启了，判断是否有签名数据，若有，根据appKey去系统中获取对应的appSecret,若存在appSecret则对请求参数进行签名，然后与传过来的签名数据进行比对) -> 检查服务方法的版本号是否过期 ->检查http请求方法是否正确，先判断调用的方法是否限制了请求方法，若有，判断请求方法与限制的是否一致，目前只支持get和post方法 -> 判断响应的消息格式是否正确，判断请求是否有指定响应格式，若有，判断是否是xml和json中的一种；检查业务顺序，依次是：应用或用户是否有访问服务的权限 -> 判断应用、会话、用户访问服务的次数限制是否超过 -> 如果是上传文件服务，检查文件类型和大小是否满足要求 -> 检查业务参数的合法性
2. ：ServiceRouter会调用ServiceMethodAdapter找到对应服务方法，然后按下面顺序执行：执行方法执行前事件 -> 服务处理前拦截,若有响应，直接返回 -> 执行对应的服务方法 -> 执行响应前拦截 -> 输出响应，ServiceRouter对得到的响应调用格式包装器(有xml和json格式，默认是xml)，得到响应消息，然后返回给客户端 -> 执行响应后事件。

客户端请求 🡪 RopServlet 🡪 ServiceRouter 🡪 SecurityManager检查系统参数 🡪 SecurityManager检查业务参数 🡪 ServiceMethodAdapter找到对应服务方法 🡪 方法执行前事件 🡪 服务处理前拦截 🡪 对应的服务方法 🡪 响应前拦截

🡪 输出响应 🡪 响应后事件