

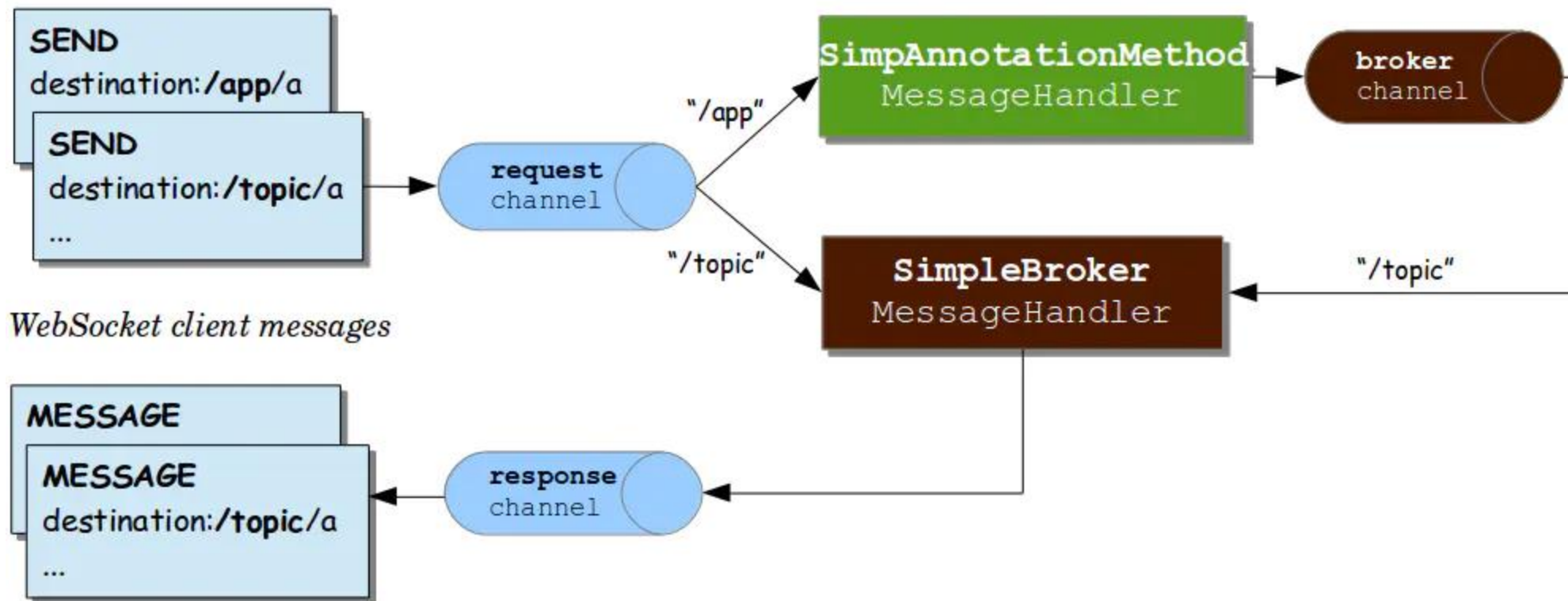
STOMP Over WebSocket有哪些好处？

使用场景	websocket	Stomp over websocket
点对点发送	消息体中内置发送到哪个用户，后台从中分离出用户，找到对方用户的session然后发送，后台需要自行维护会话池（下同）	用户信息放到destination(类似于队列的topic), 消息体本身不用与用户id耦合，基于topic编码更直观、简单
广播发送	同样要内置到消息体，与后端约定好标识（如“all”），后台收该标识的消息，则需要遍历所有session会话并发送	过程同上面类似，后台只需用注解SendAll即可，不需要管理会话session
业务主题	不基于用户，而是主题情况，如报警提醒、订单提醒等，并不是按人来划分的，实现需要后台维护好topic和用户的关系，实现非常复杂	直接基于消息队列topic，发布订阅模式
负载均衡	实例之间相互跳转，复杂度高、性能差	有基于第三方的Stomp代理，几乎不需要改造代码即可实现
兼容性	客户端（浏览器）必须支持websocket协议	可以降级使用长轮询等方式

STOMP Over WebSocket高级特性

使用场景	websocket	Stomp over websocket
心跳检测	自己编码实现	内置实现，只需要做配置即可
消息确认、重发	自己编码实现	内置实现，采用ACK、NACK、RECEIPT
事务控制	自己编码实现	内置实现，采用BEGIN" "COMMIT" "ABORT"

STOMP Over WebSocket 通讯流程



独立Stomp代理（如RabbitMQ）

