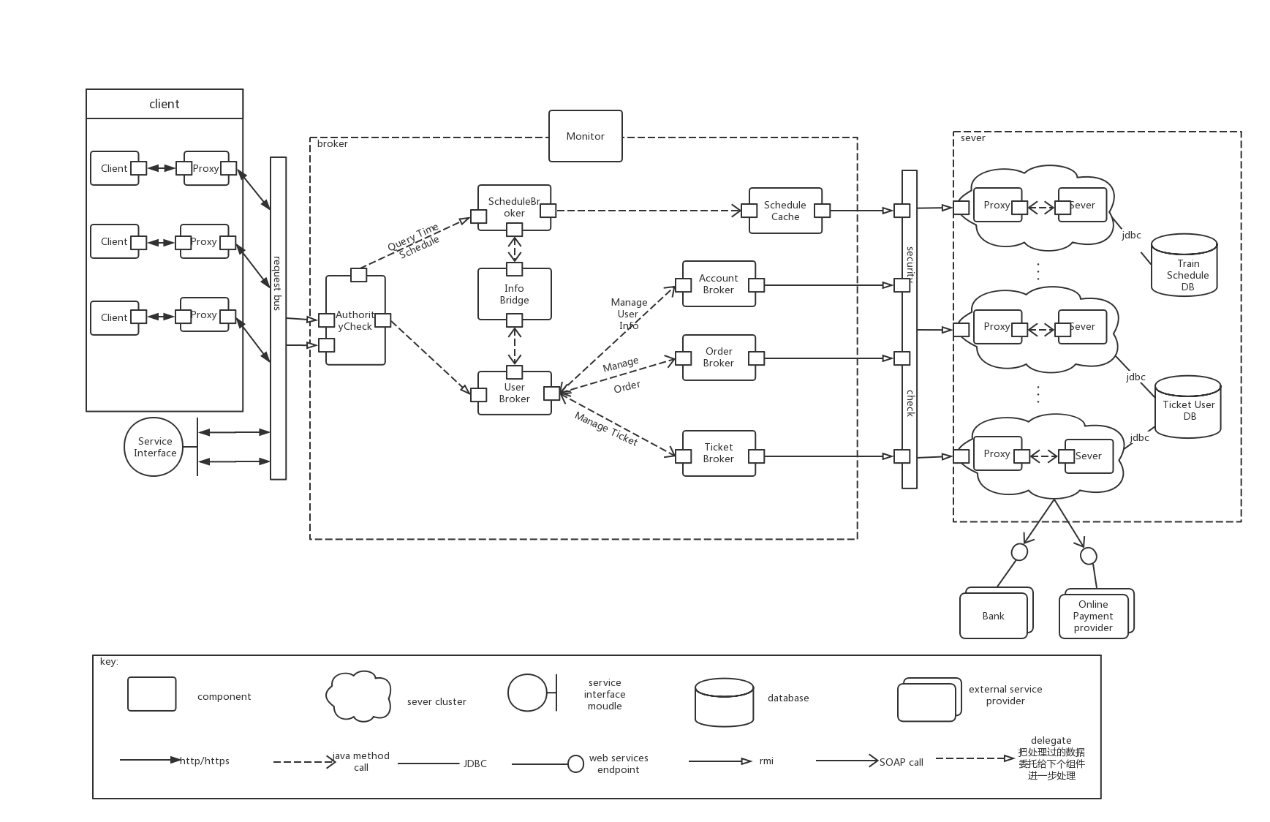
### C&C视角

#### 视图的主要表示（分解视图）



#### 元素目录

本架构为Broker架构，主要元素由client、client proxy、broker、server proxy、server组成以及之间的Connector组成。

图中request bus负责客户端请求的分派处理，可以识别请求类型并为其指派不同的broker进行处理，分派完成之后请求会先发送给AuthorityCheck模块进行安全验证和数据加密（主要包括拦截dos攻击和非法请求），之后再通过它和broker之间的接口分发给指定好的broker模块；monitor为系统对broker整个模块的监控器，负责周期性发送heartbeat对broker的健康状况进行判断，并处理异常情况，同时它也可以根据请求响应时间来对broker等待任务集进行动态调度，可以降低broker任务阻塞的可能性；broker任务转发完成之后通过和security check之间的接口将请求转发给对应的服务器， security check负责对broker和服务器之间传递的信息进行加密以防止信息泄露；Authority check和security check这两个安全验证模块加强了broker的入口和出口的安全性，有效的降低了broker单点失效的可能性。

在request bus将不同的请求识别并分派给不同的broker之后，schedule broker负责处理有关列车调度时刻表等信息的请求，user broker负责处理用户账户、订单以及车票管理等信息的请求，由于user broker负责的功能较多，又对user broker的请求进行了二次分派，实现了负载均衡，具体信息可以查看broker部分的模块分解；server端负责客户端请求数据的查找与存储，以及系统对外提供的接口，该模块在内部也进行了分解提高了性能、可用性、安全性等质量属性，具体分解可以查看server端的模块分解。

元素的接口和行为：

|  |  |  |
| --- | --- | --- |
| 接口名称 | 接口职责 | 接口类别 |
| 客户端请求分派接口 | 给内部requestbus进行请求分派 | 内部接口 |
| Request bus请求分派接口 | 给内部权限验证模块先进行安全验证和数据加密，并提前指定每个请求的分派broker | 内部接口 |
| Authority安全模块接口 | 将通过安全验证的请求分发给user broker和schedule broker | 内部接口 |
| 列车调度broker接口 | 将列车调度相关请求先发送给服务器安全模块进行处理，并提前指定提供列车时刻表检索服务的server端 | 内部接口 |
| 用户broker接口 | 将到达的请求分发给user account broker、order broker、ticket broker进行二次分派 | 内部接口 |
| 用户账户broker接口 | 将用户账户相关请求先发送给服务器安全验证模块进行验证，并提前指定所要分发得提供对应服务的server端 | 内部接口 |
| 订单broker接口 | 作用同上，只是它操作的是订单相关请求 | 内部接口 |
| 票务broker接口 | 作用同上，只是它操作的是票务相关请求 | 内部接口 |
| Security check服务器安全接口 | 将从broker拿到的信息进行加密处理，并发送给对应的server | 内部接口 |
| server端服务接口 | 提供银行相关信息和在线支付订单的服务 | 外部接口 |

#### 可变性指南

系统的可变性主要体现在：

1. broker架构中的client和server端可能会经常发生变化，但是他们不依赖于彼此，他们只依赖于broker，这个设计使得他们可以通过在broker注册和解注册进行灵活地修改和扩展
2. broker在运行时有monitor对状态进行周期性监测，如果出现崩溃或异常，可以快速利用冗余组件恢复broker状态。
3. server端和request bus都有heartbeat模块进行周期性监测，也有冗余处理，可以快速处理系统的异常，提高系统可用性
4. 安全模块在运行时对客户端请求进行验证加密，可以通过请求次数验证等方式防御dos攻击，也可以通过记录用户登录次数进行用户访问权限的控制，大大提高了系统的安全性
5. Request bus对请求进行智能分派和排队调度，可以避免个别请求占用系统资源而造成阻塞调度的情况发生
6. 分布式数据处理存储，可以方便的增加或者删除server端