

Ejabberd With MQTT

simpcl

2015.10.30

Ejabberd

- 开源 IM 服务器
- 使用XMPP协议组织消息，可以基于TCP或HTTP进行通信
- Erlang语言开发，稳定，可支持巨大客户端连接

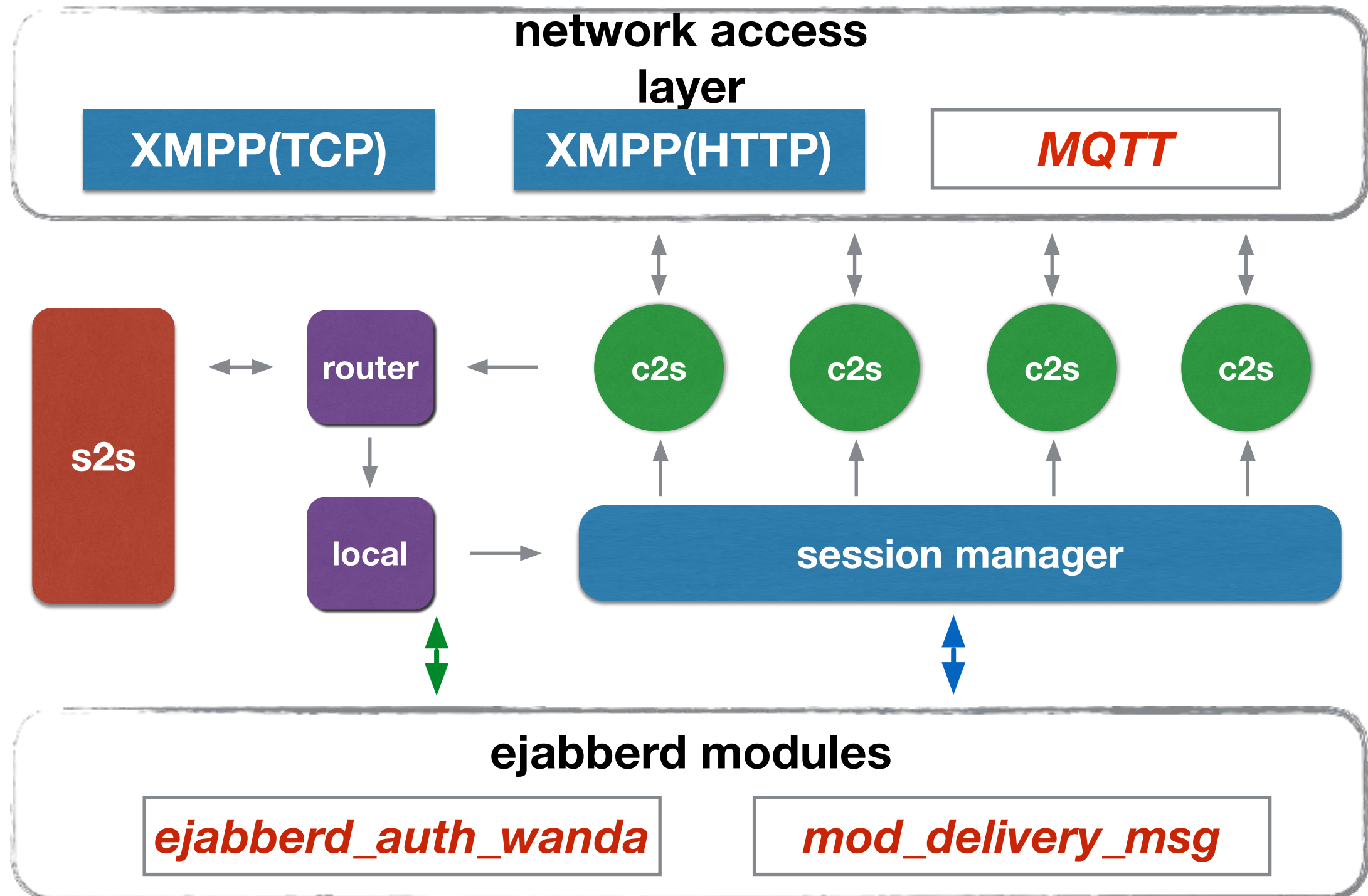
XMPP 客户端

- API
 - Java - Smack
 - Object-C - XMPPFramework
 - Python - SleekXMPP
- Software
 - Windows - Pidgin
 - Mac - Adium
 - Linux - Pidgin

New Ejabberd !

- 集成会员验证功能
- 支持MQTT轻量级协议
- 兼容已有的消息推送
- 定时(延时)消息功能

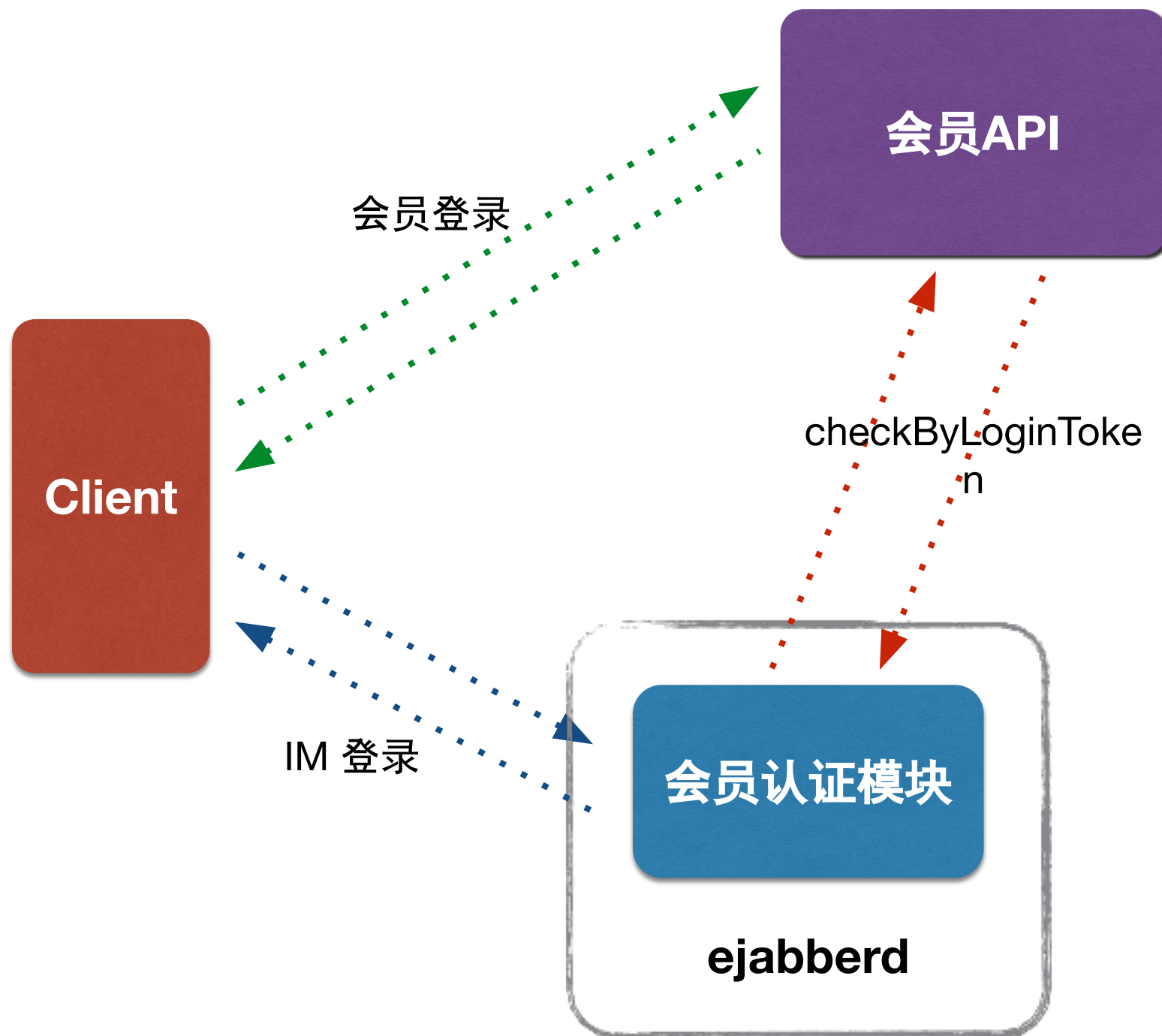
New Ejabberd !



会员验证

- XMPP协议使用JID(username@domain/resource)作为用户的唯一标示
- 在会员登录之后都具有uid和loginToken
- uid为用户名, 即JID为: uid@domain/resource; loginToken为密码, 无需注册, 可使用任意客户端软件或API即可登录IM服务器
- Ejabberd的会员验证模块会获取uid和loginToken, 调用会员API进行验证

会员验证



MQTT

- 简化XMPP登录过程客户端与服务器的多次交互
- 减少额外的数据传输
- 客户端API使用更简单, 降低客户端开发复杂度

MQTT

- MQTT协议, client_id为客户端唯一标示, username与password可选
- 为兼容XMPP协议, 使用MQTT协议登录ejabberd, resource为client_id, uid为username, loginToken为password
- publish操作的topic为消息发送目标联系人的uid
- 默认订阅以自己uid为名称的topic, 无需做subscribe操作, 即可收到发给自己的消息

MQTT

发送消息:

```
publish.single('ut001',  
  
    'Hi, ut001',  
  
    hostname='10.77.144.249',  
  
    port=10883,  
  
    client_id='chendemac-pro',  
  
    auth={'username':'ut003', 'password':'123'})
```

MQTT

接收消息:

```
def mqtt_on_message_callback(mqttc, obj, msg):
```

```
    print 'recv message:', msg
```

```
if __name__ == '__main__':
```

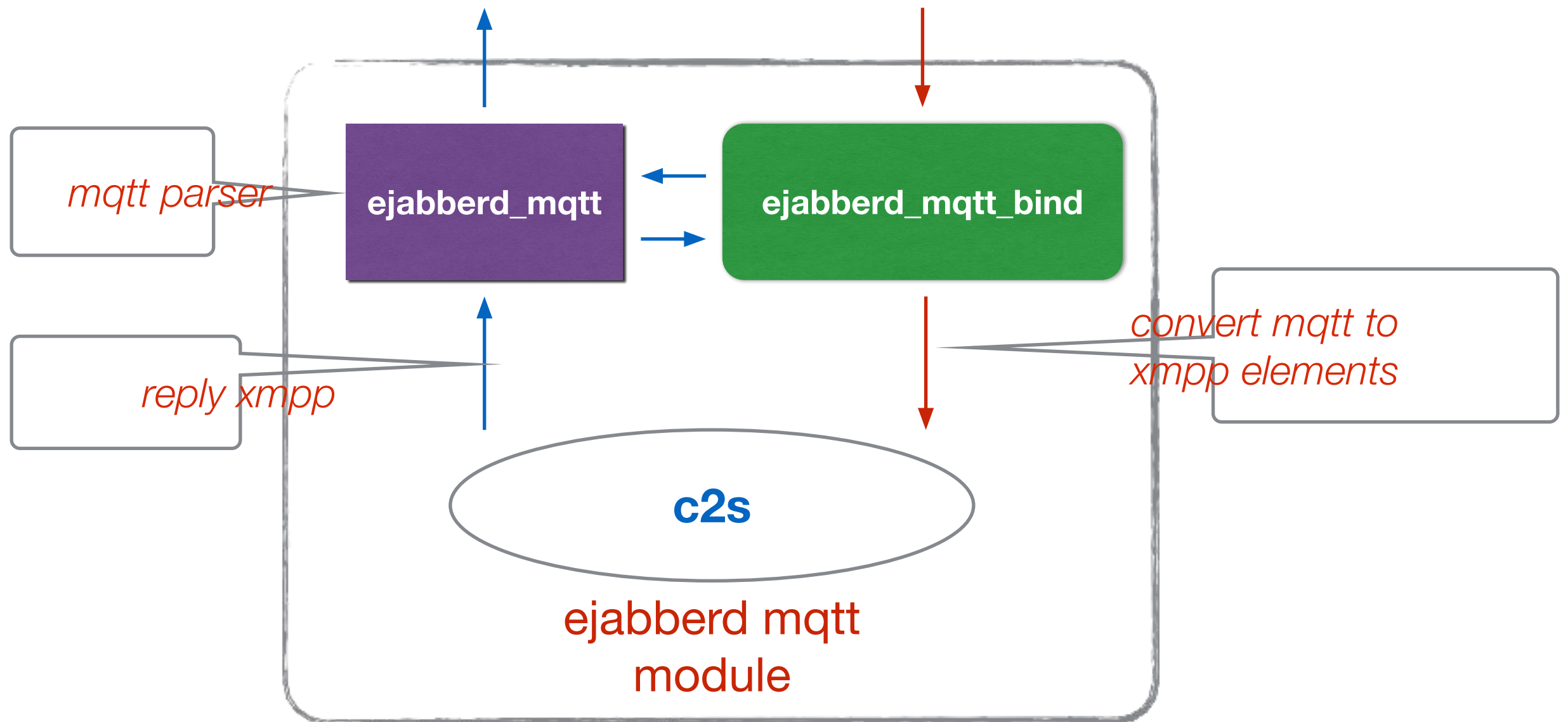
```
    mqttc = mqtt.Client('chendemac-pro') // new with client_id
```

```
    mqttc.username_pw_set('ut003', '123') // set username and password
```

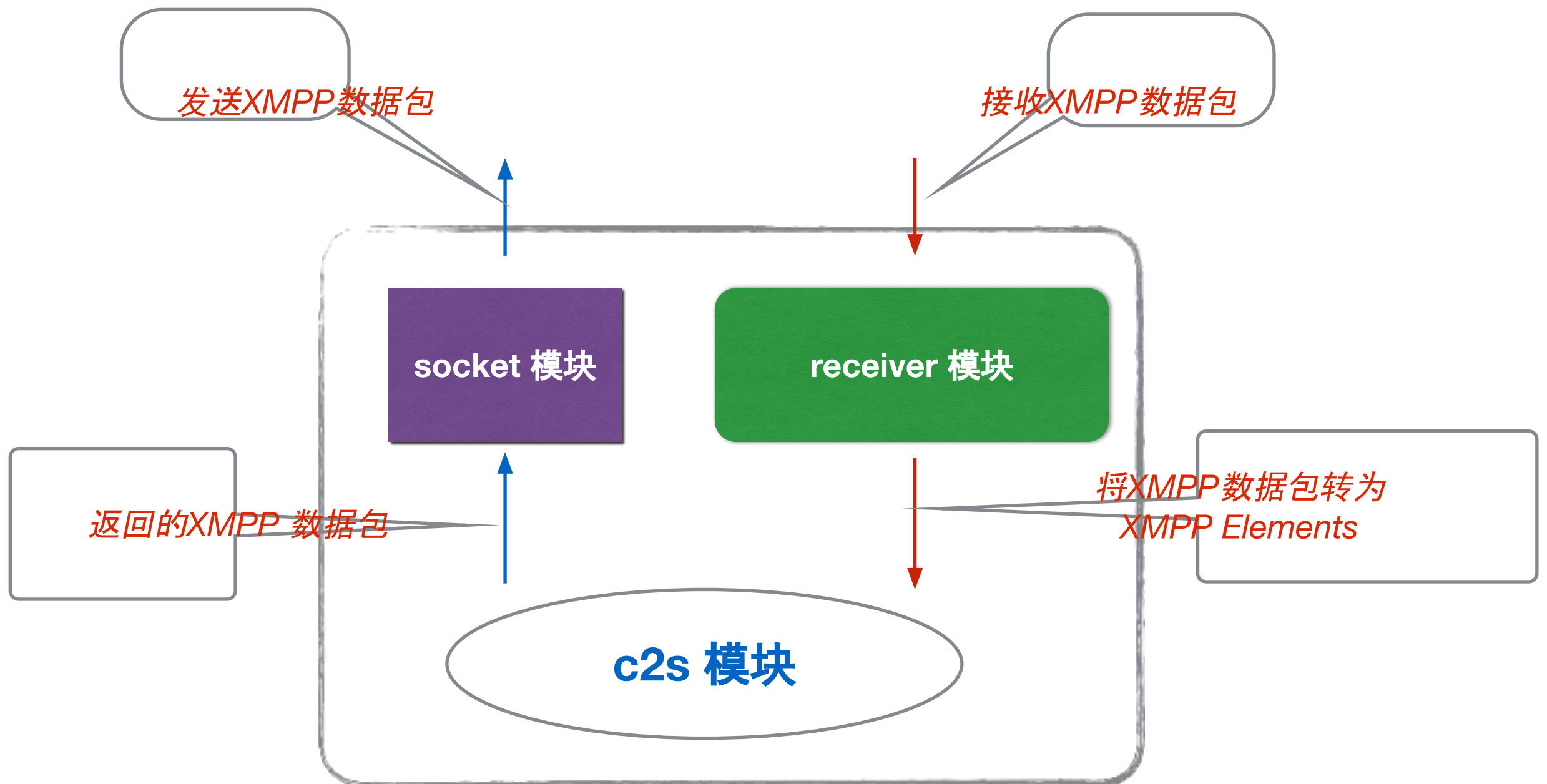
```
    mqttc.on_message = mqtt_on_message_callback
```

```
    mqttc.connect('10.77.144.249', 10883, 60)
```

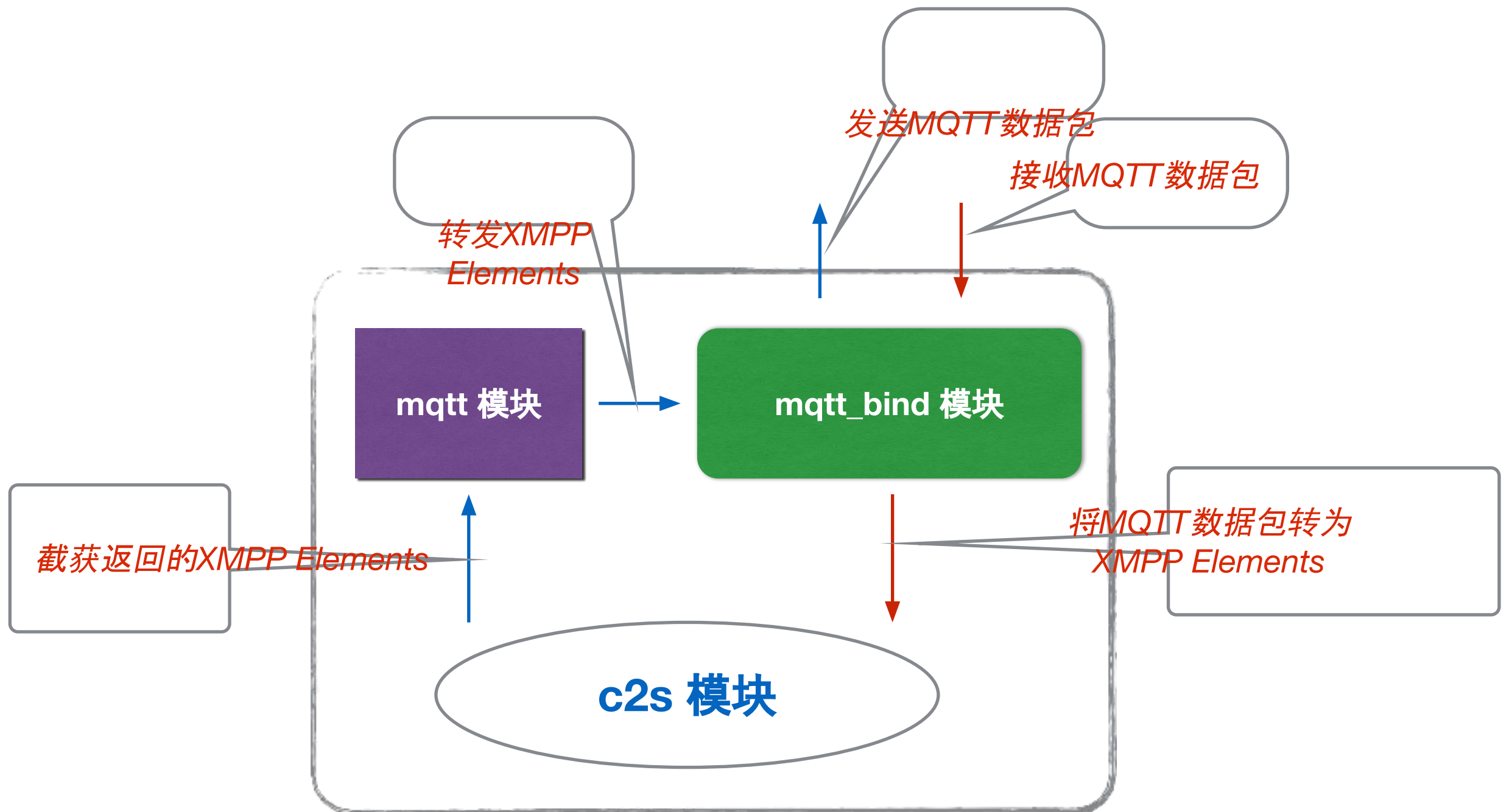
MQTT



MQTT



MQTT



MQTT

- 包含ejabberd_mqtt和ejabberd_mqtt_bind
- ejabberd_mqtt: 负责MQTT协议解析
- ejabberd_mqtt_bind: 负责与客户端的网络通信
- ejabberd_mqtt_bind: 将客户端MQTT协议转换为XMPP的Elements, 与ejabberd_c2s通信

消息推送

- 消息推送

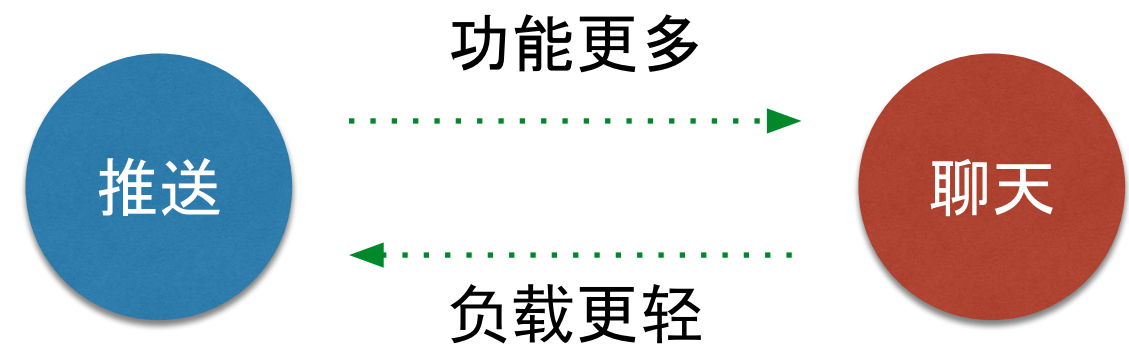
1. 按Topic订阅消息

2. 只收不发，单向

- 聊天

1. 按用户ID接收消息

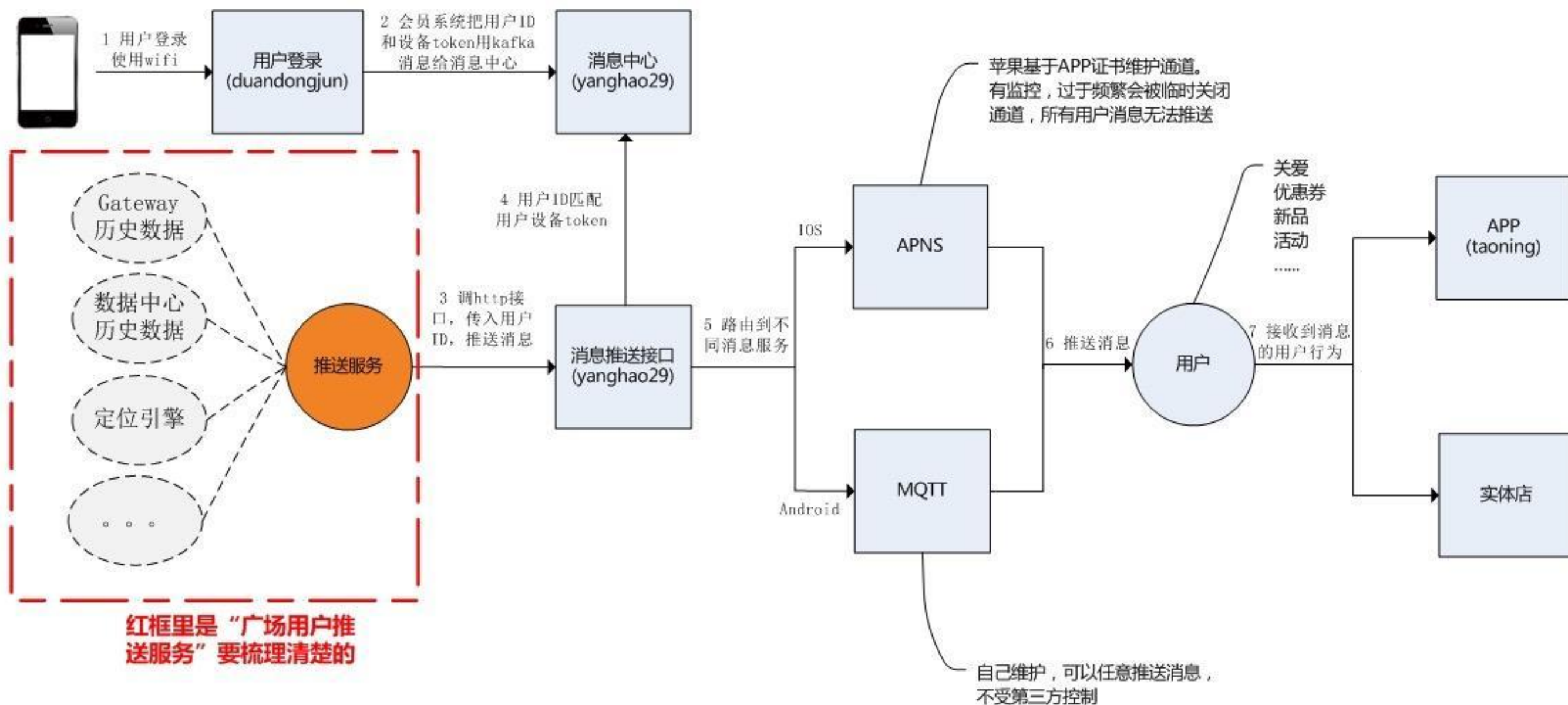
2. 可收可发，双向



消息推送

今天找应用那边了解了消息推送的现状:

- 1, 目前已经有消息推送服务, 由运营人员触发推送, 没有系统自动的推送
- 2, 目前几种消息模板, 例如优惠券、商品详情, 用户收到这些消息, 点击会跳转到app站内, 如果我们这次不新增消息模板, app端不涉及开发工作量, 如果新增消息模板taoning可以支持
- 3, 消息推送对广场内的商管影响大, 需要先跟运营方沟通



消息推送

- 手机会员登录
- 手机请求消息中心HTTP接口, 分配mqtt server的ip+port
- 手机连接mqtt server, 请求参数:

clientId=udid(device id)

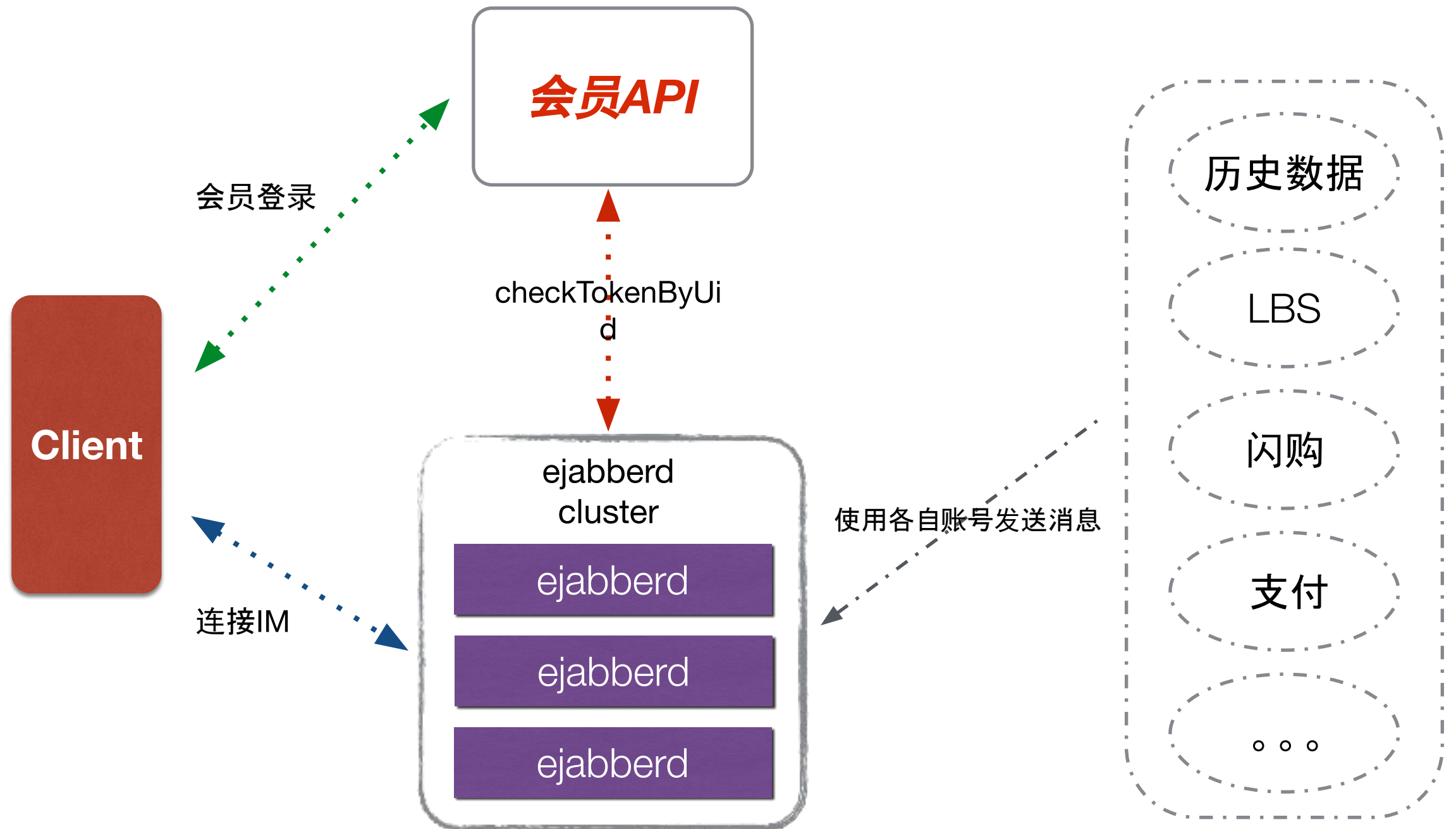
userName = uid

password = sessionid

消息推送

- mosquitto, 部署多个实例, NAT映射, 占用多个端口
- 每增加一个推送的业务类型, 需要推送服务端实现增加一个 detailld
- 一个用户只能推送一个设备
- 1 user <-> 1 topic, 聊天场景

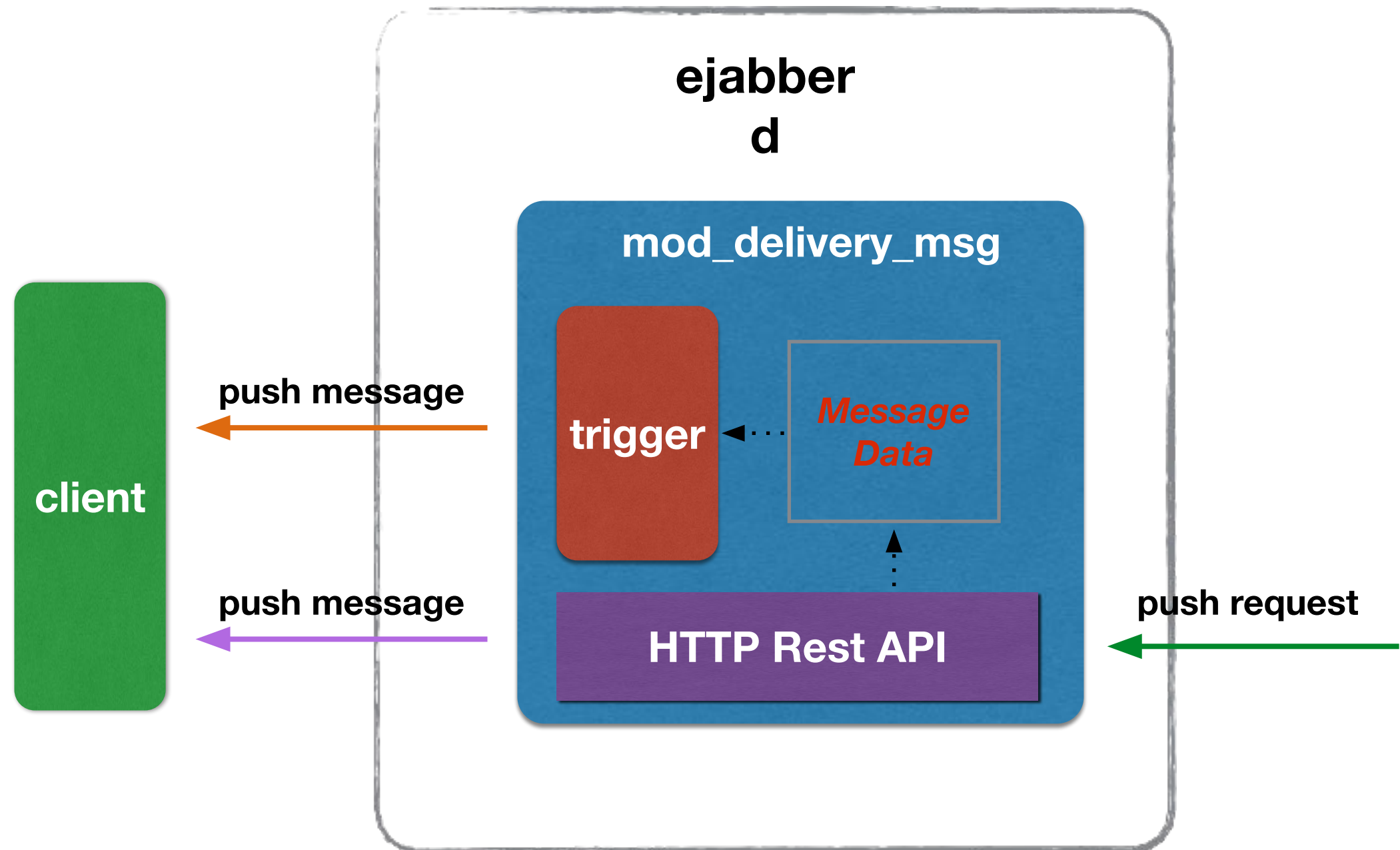
消息推送



mod_delivery_msg

- 定时(延时)发送消息
- 定时发送的消息在发送之前可更新
- 作为独立项目imps已经上线, 应用到闪购提醒
- 考虑集成到ejabberd中

mod_delivery_msg



mod_delivery_msg

- 包含http rest api和trigger
- http rest api负责接收推送消息的http请求，没有发送时间要求则立即发送
- trigger负责触发发送动作

未来的事情

1. 消息推送替换为ejabberd+mqtt
2. ejabberd_mqtt module refactor, 不依赖c2s
3. 支持QoS
4. IM业务场景

Thanks.

Q & A