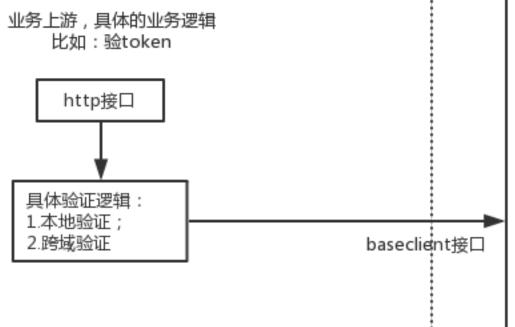


域A

域B



对外封装一个函数,

call_cross_region(module_name, method_name, objectid, payload, regions = None);

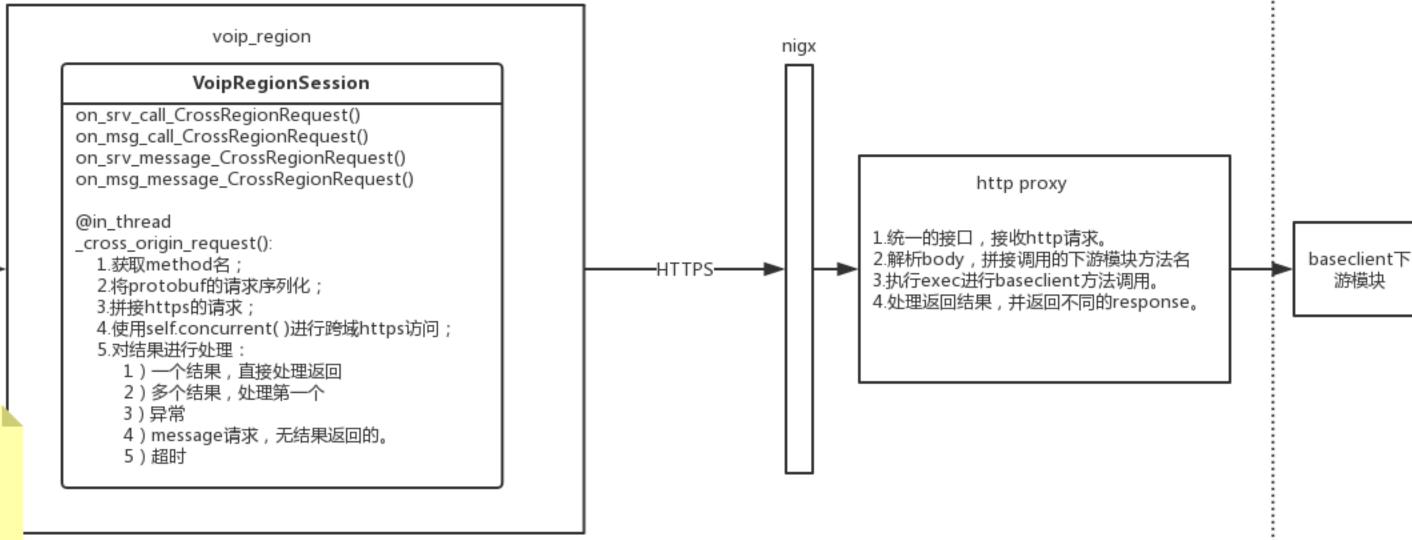
module_name: 服务名

method_name:

service_call/object_call/service_message/object_message objecid = None, 如果是service服务, 无需赋值。

payload: proto格式的请求。

regions = None, 一个List, 请求访问的域。默认None时, 对 所有的域进行广播请求。



```
https请求的body, 定义为json字段:
   'req_id': 域名_自增数字。(多实例,id可能会重复)
  'from_region': 请求的域名
  'to_region':
  'module_name': 'VOIP_CHARGING',
  'method': 'serve_call'/'object_call'/'service_message'/'object_message',
  'object id':
  'request_body': photo 序列化后的字符串
   'timestamp': 时间戳,统一使用UTC时间
返回的response字段:
  'rsp_id': 域名_自增数字_响应域
  'from_region':
   'to_region' : 响应的域名
  'module_name'
  'method': 'serve_call'/'object_call'/'service_message'/'object_message',
  'error_code': 0
  'error_msg': 'OK'
  'result': response序列化后的字符串
   'result_type':结果的类型
   'timestamp':时间戳,统一使用UTC时间
  'handle_ip': 处理请求的ip
  'handle_instance': 处理请求的实例号
```