

MICROOH 麦可网

Android-从程序员到架构师之路

出品人：Sundy

讲师：高焕堂（台湾）

<http://www.microoh.com>

F05_a

观摩：Android端云整合 与分工策略(a)

By 高煥堂

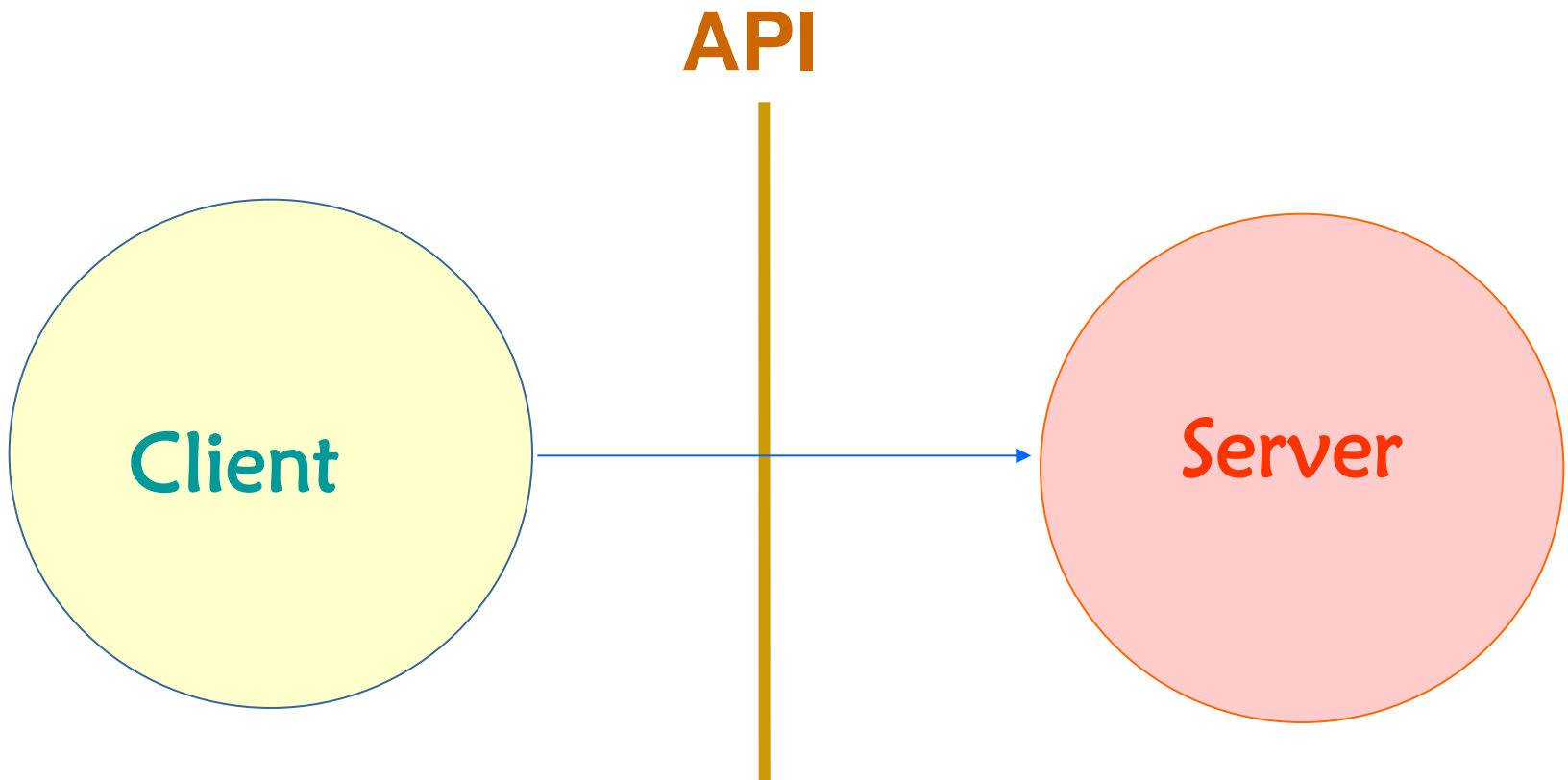
内容

1. API：架构师分工决策的呈现
2. GAE云平台的API
3. 在GAE云平台上使用Proxy-Stub设计模式

1、API： 架构师分工决策的呈现

古典API，传统分工

- 古典Client/Server架构的API呈现于Client与Server之间，成为两端分工生产(或开发)的界线。
- 这种「古典API，传统分工」模式，如下：

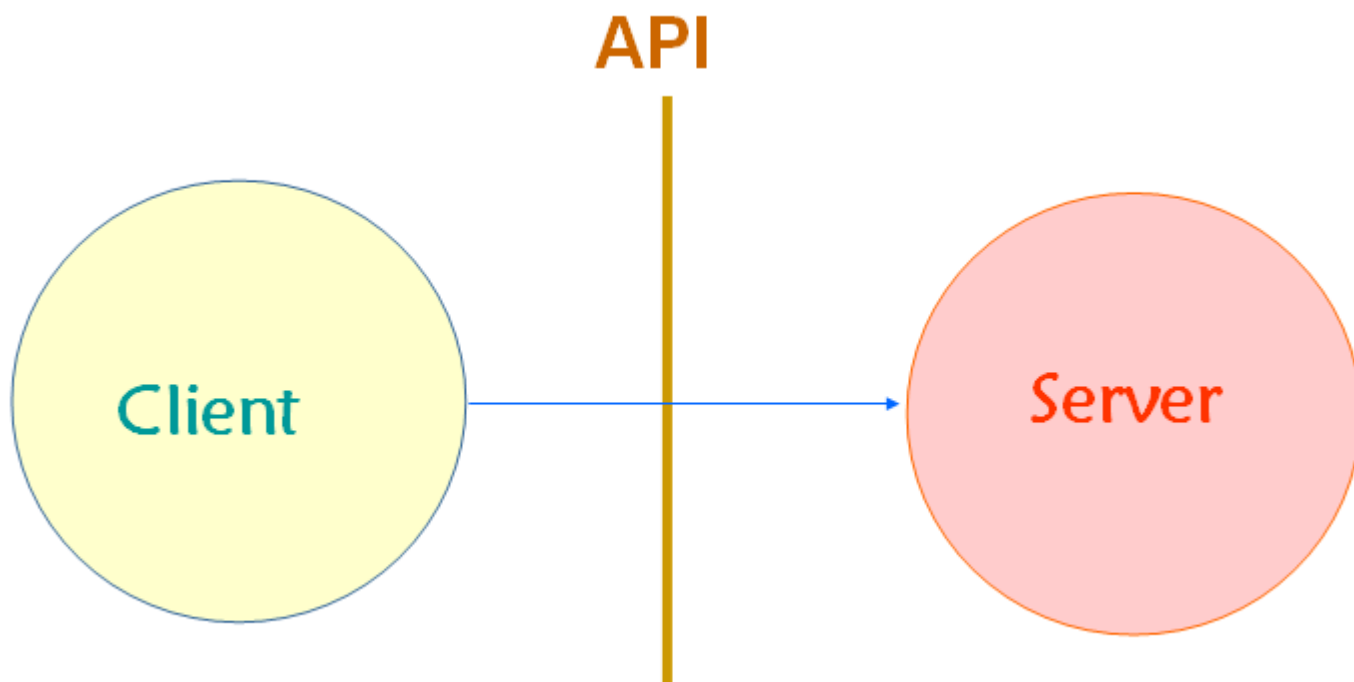


- 这种API不利于Server端团队(或企业)。
- 对于Server端团队而言，这是被动型API，缺乏主控权，经常沦为Client端所指挥的小弟或小妹，而成为救火队。

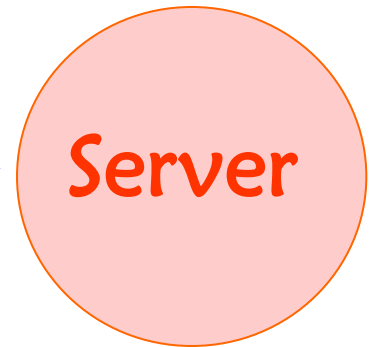
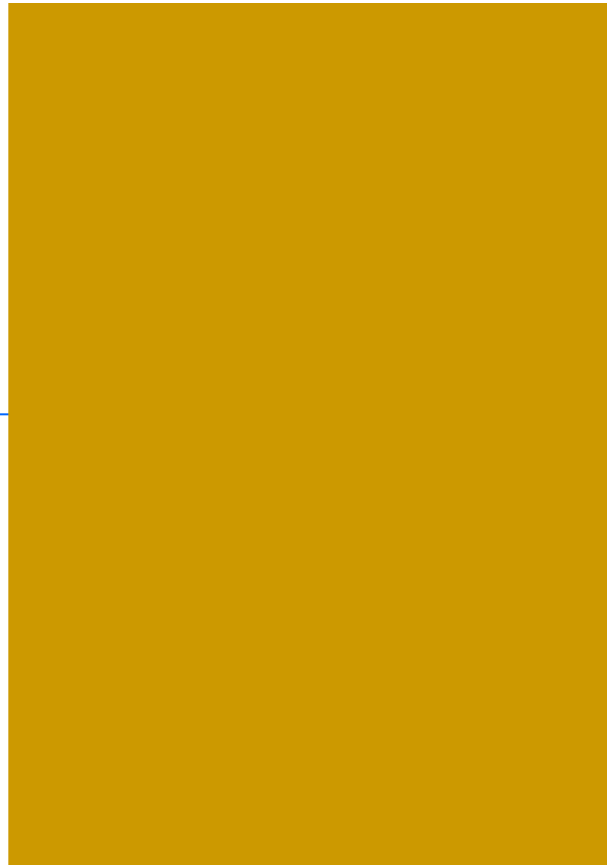
- 如果Server端企业想成为<强龙>，就必须想办法掌握API的话语权。

架构师如何重新定位API呢？

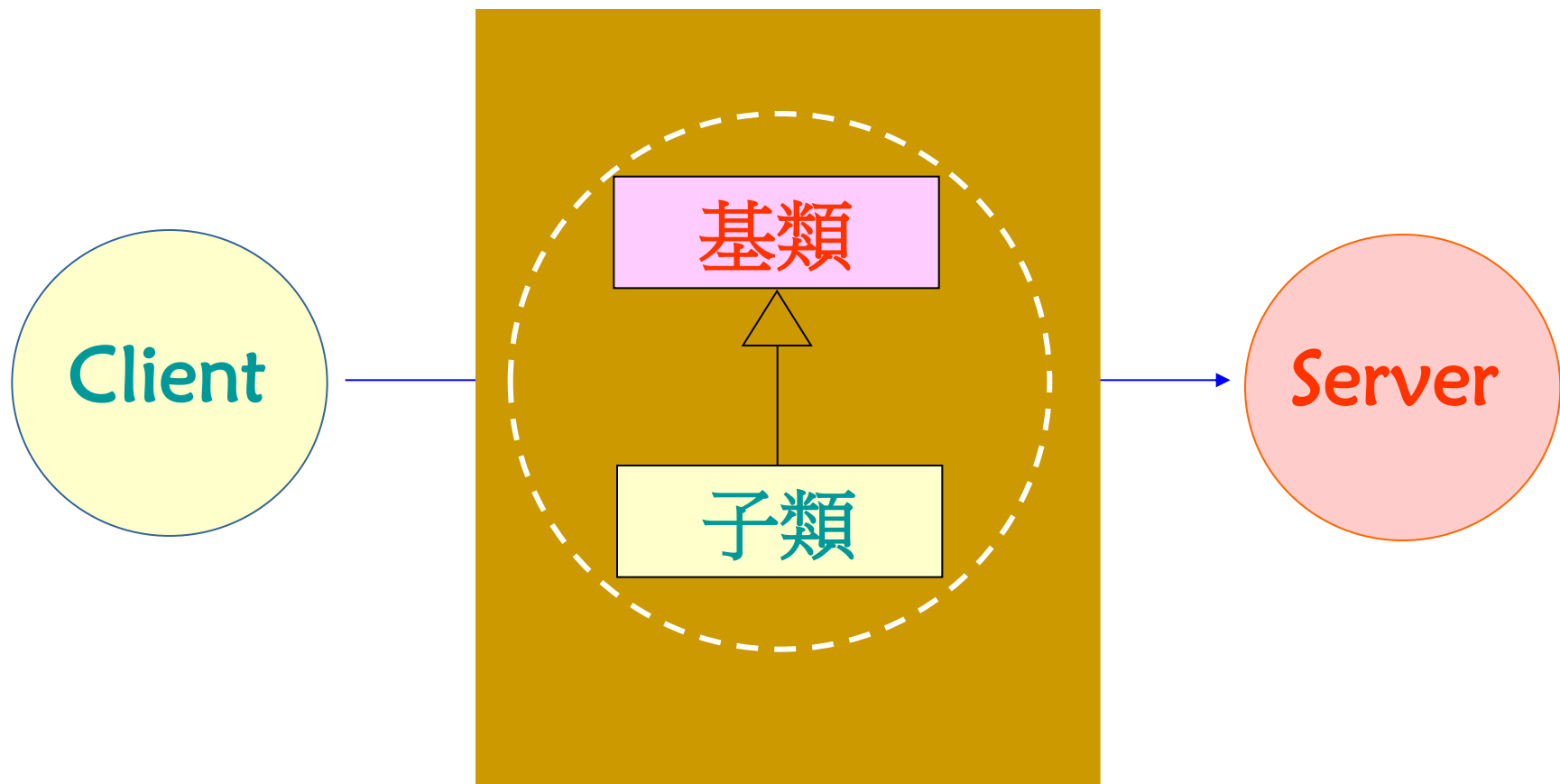
重新定位API



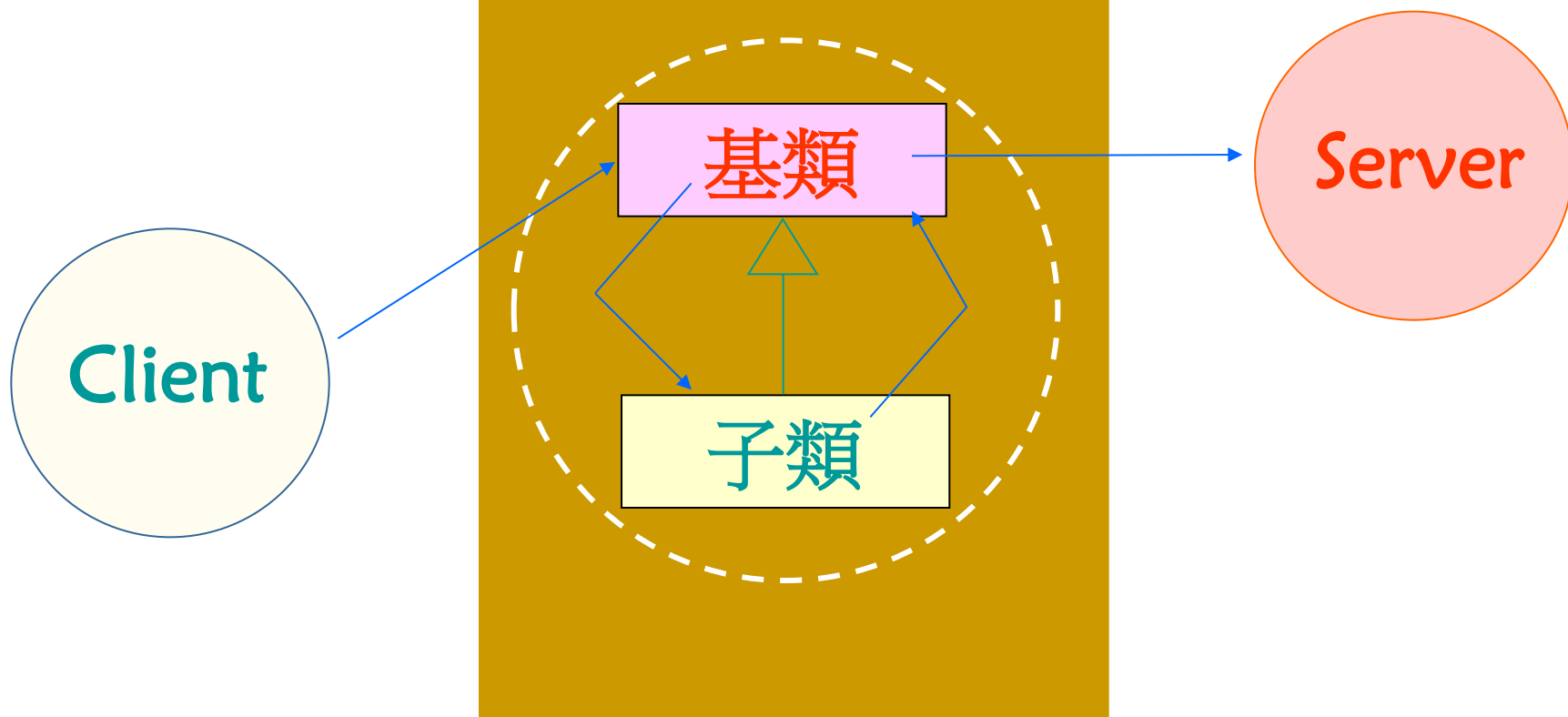
API

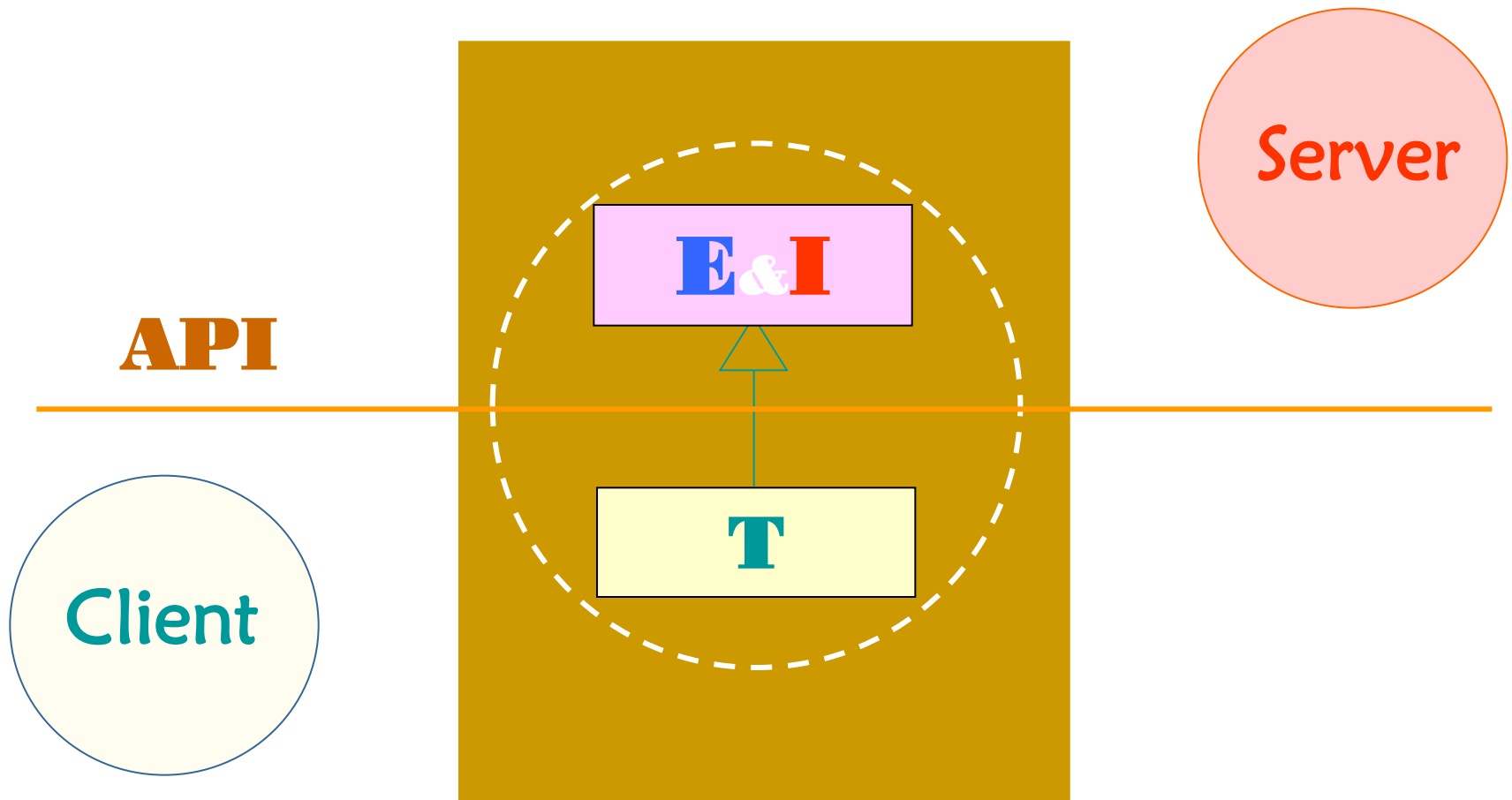


API

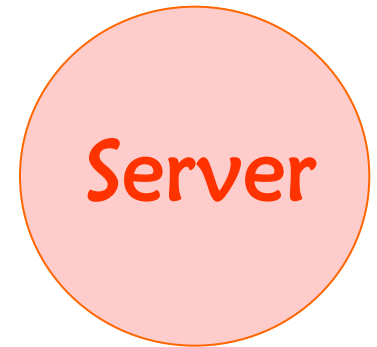
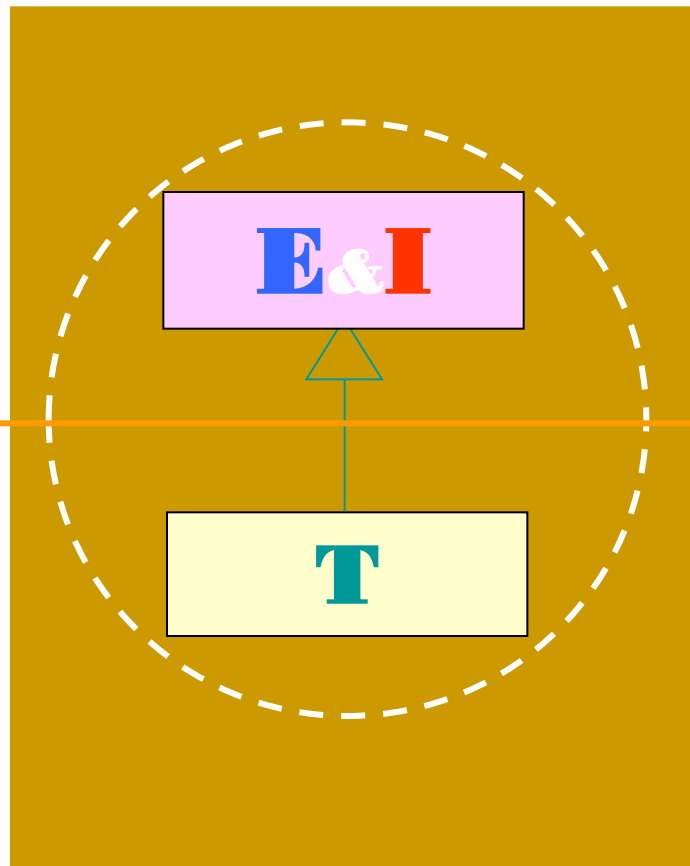


API





API = I

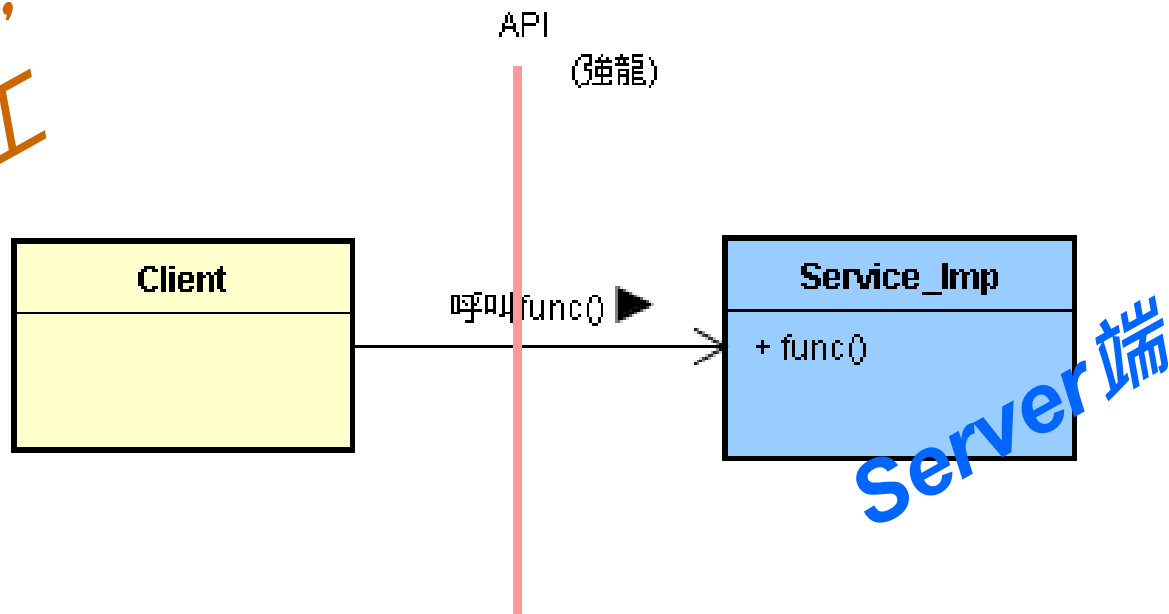


EIT造形放在Server端，
还是Client端呢？

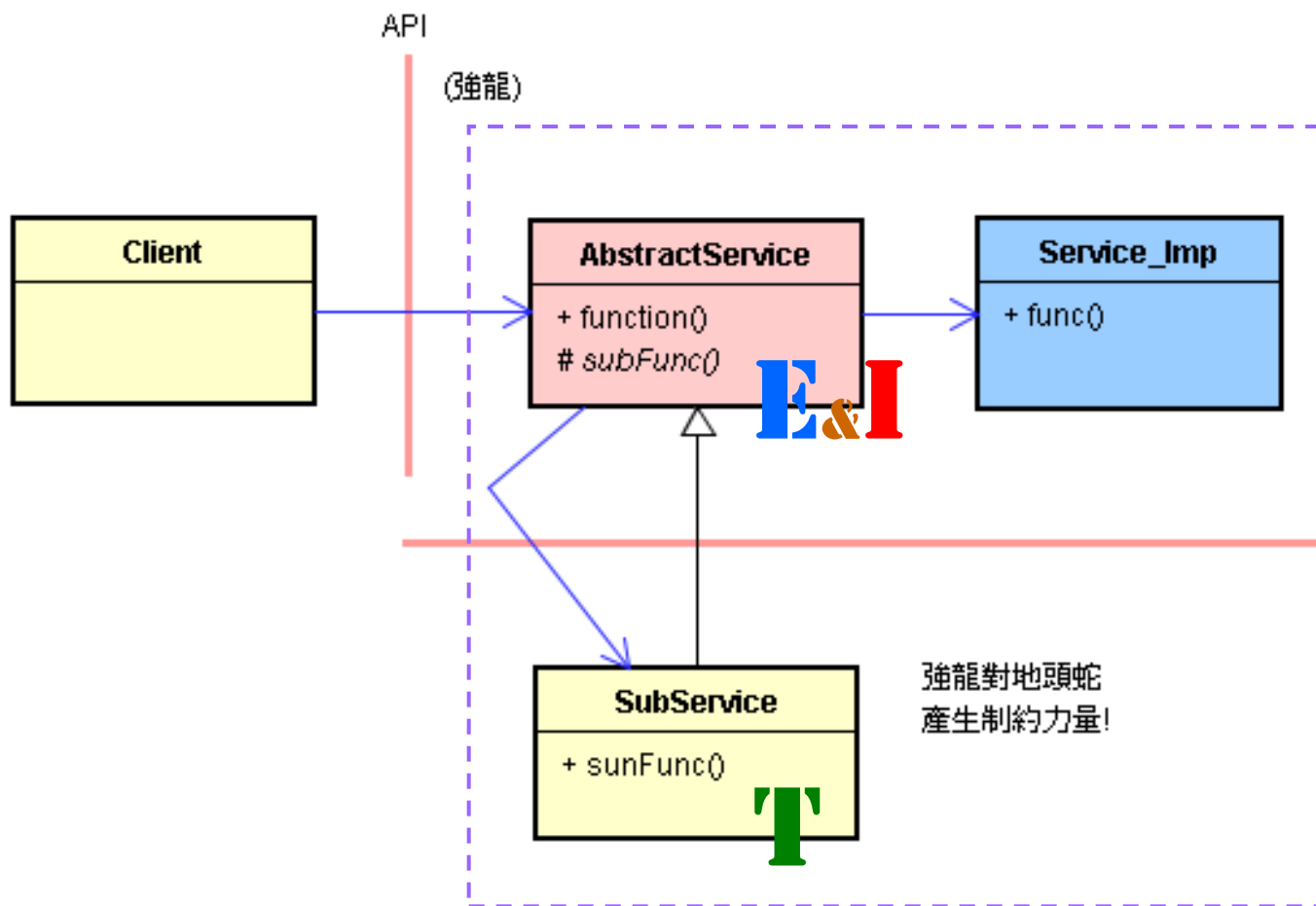
答案是：两边都放，
两边同步改变API定位、
改变分工界线

将EIT造形运用于服务端

古典的API，
傳統的分工

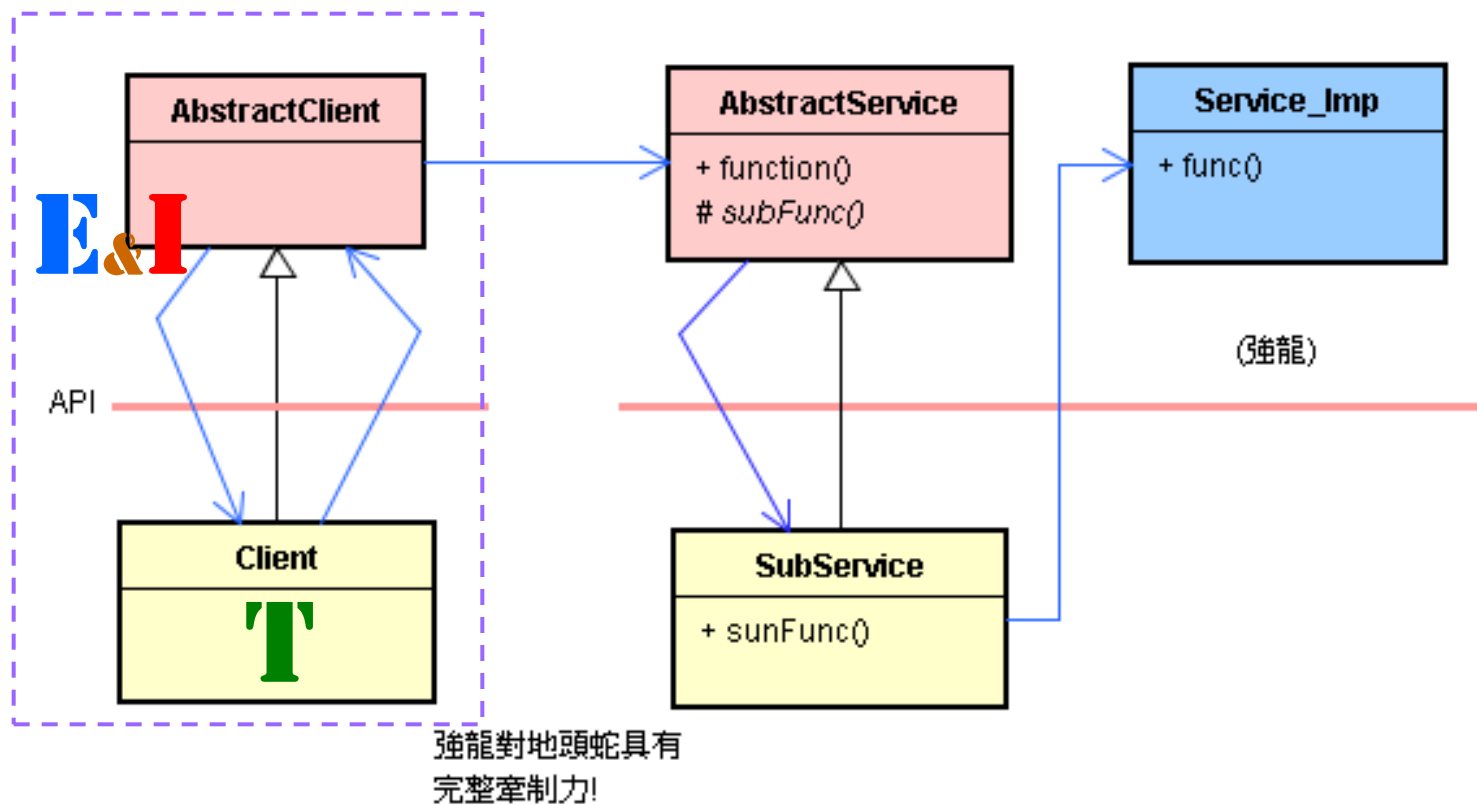


- 其中，Service_Imp提供服务给Client，使得Server端受制于Client端，Server端对Client端没有主导力量。于是，Server端常常成为救火队而疲于奔命。
- 此时，Server端开发自己的<E&I>，来提升自己的主导力量，就不必再疲于奔命了。如下图：

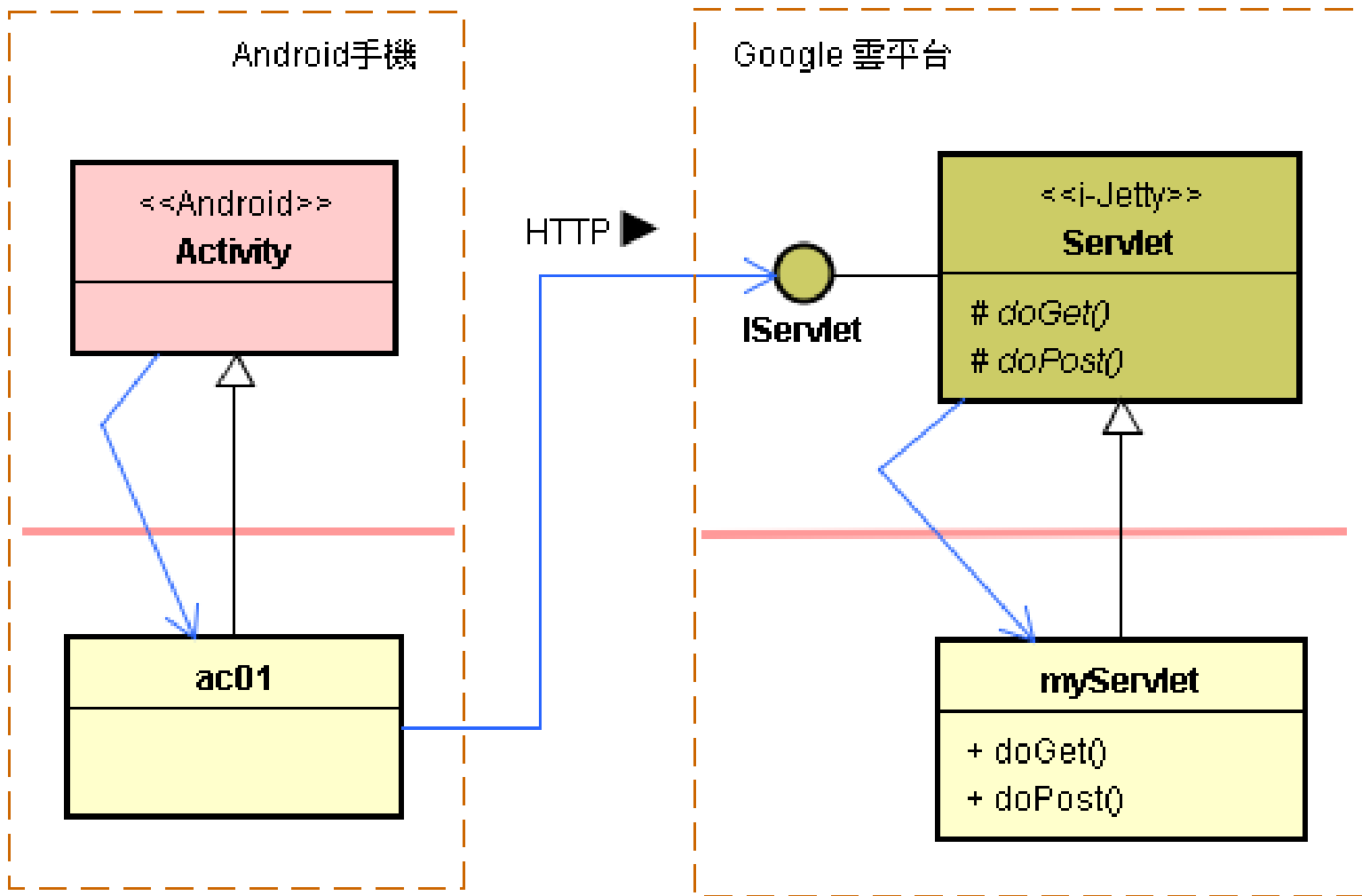


- Server端获得了更大的主导权，就会大胆地开放子类别给更多的Client端人员去开发了。随着Client端的数量愈多，其地位就愈高，日益成为云平台的强龙了。
- 成为云平台的强龙了，将同样策略应用到终端上，顺势征服终端，就能为真命天子：云(大)强龙了。

将EIT造形运用于终端(Client)



- 无论是在Server端，还是Client端，取得全面性的主导权，因而大胆地在两端都开放出插件。
- 例如，Google把这种策略应用到它的GAE云端和Android手机终端上。



- 应用于云端向终端的信息推送，例如股票分析、车联网服务；以及手机多人游戏等等。

股票信息雲



股票分析師



車載雲



推送股市
分析信息

推送貨車
位置信息



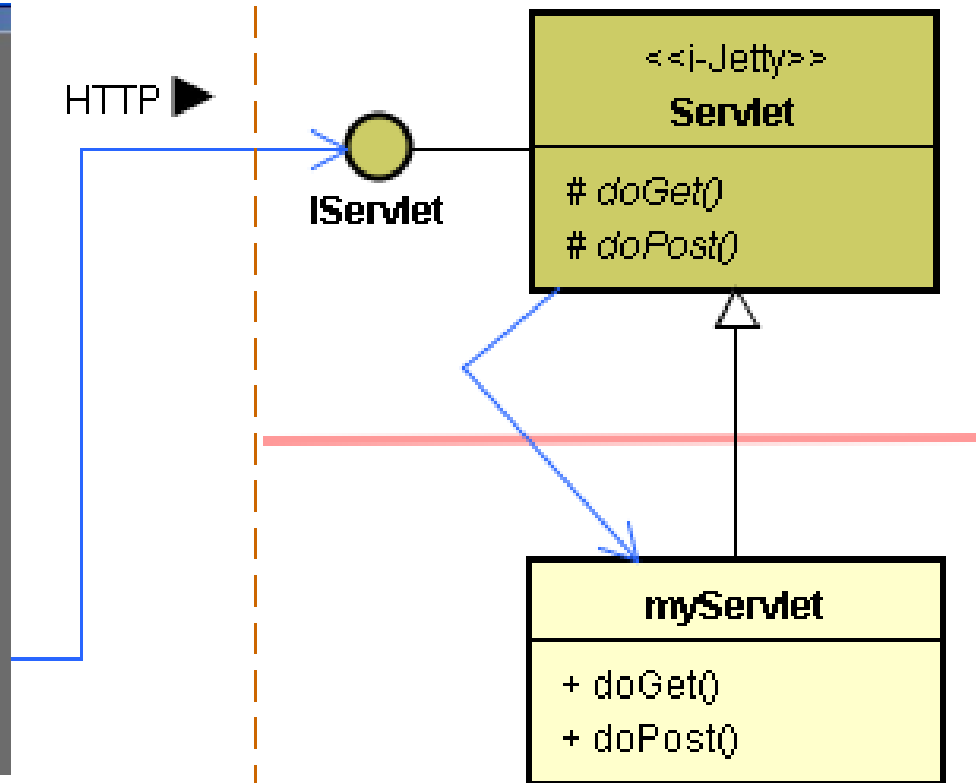
智慧
終端

Android手機

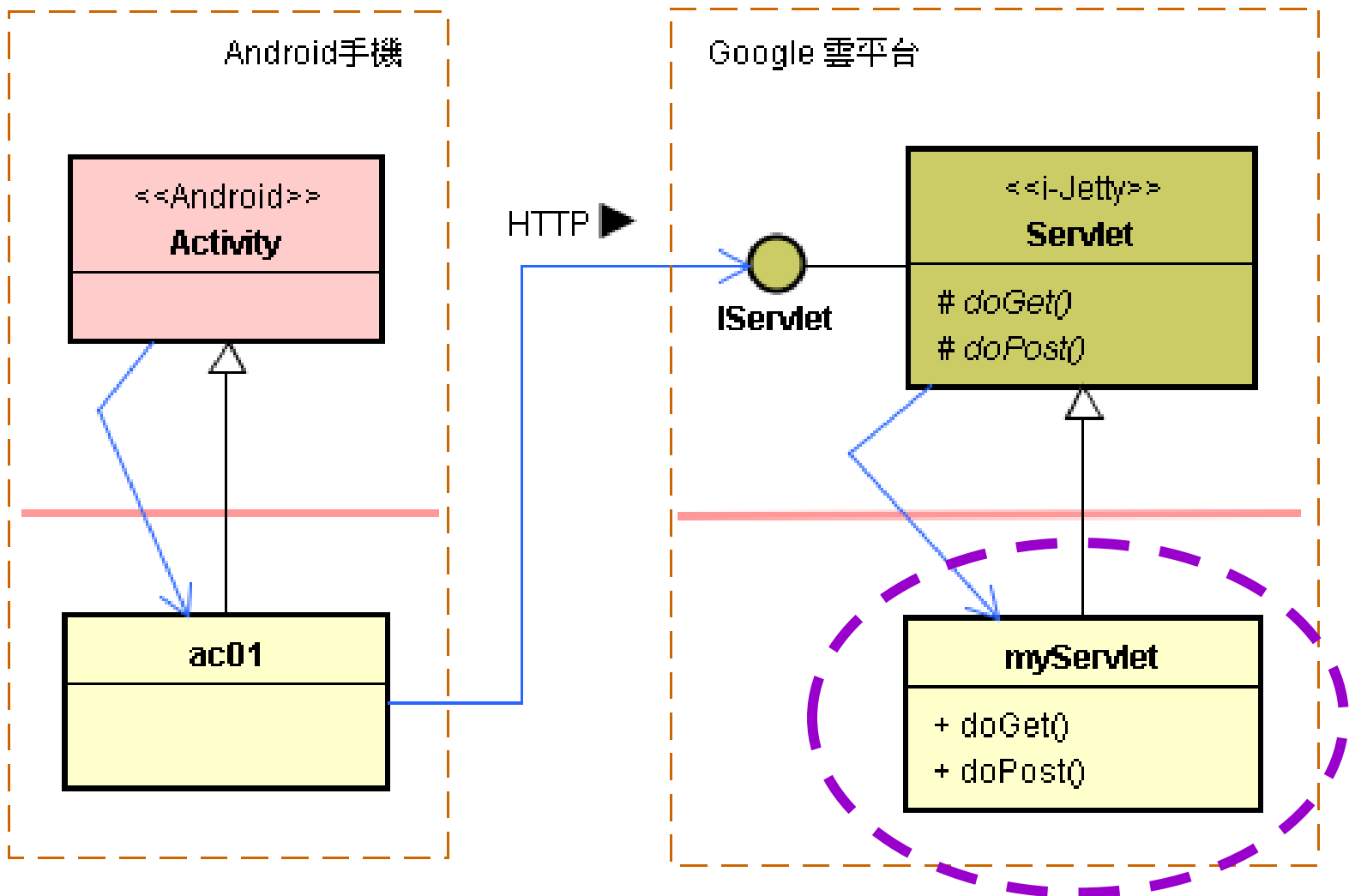


Google 雲平台

HTTP



- 基于新的API定位(也是新的分工界线)，强龙在两端都写<E&I>；而地头蛇在两端都写<T>。
- 待会儿，将看看如何写Google的云端(GAE)上的<T>代码了。





~ Continued ~