链滴

初见微服务之RESTful API

作者：2440407207

原文链接：https://ld246.com/article/1461644033676

来源网站：链滴

1. REST名称由来

REST全称为Representational State Transfer，即表述性状态转移，最早由Roy Feilding博士在世纪交（2000年）提出，喜欢追根溯源的朋友可以读一下他的博士论文《Architectural Styles and the D sign of Network-based Software Architectures》，这时距HTTP1.1协议标准正式发布（1999年6）仅一年的时间。

岁月的痕迹跨越了十多年，技术的进步日新月异，所有的人都在谈论着应用容器化、服务解耦、DevOs开发运维文化等等。我们变得喜新厌旧，技术成了快餐，框架是越来越多的舶来品。此时，我们是应该静一静，看看技术的起源，想想我们如何成为软件的设计师，而不是代码的奴隶、资本的工具？RST作为历史的宝藏，被越来越多的人挖掘、归纳、推陈出新，近几年占领了几乎所有的大型互联网公的开放API，国外如google（https://developers.google.com/apis-explorer）、facebook，国内有豆瓣、腾讯的公众平台等。

在这里，我要替SOAP说几句话，技术的进步始终是从无到有，由繁入简的。在一定的时间里SOAP足了web服务的设计要求，达到了对外提供服务的目的，尽管十分的（协议）晦涩、（解析）生硬。业级的软件依然有很多保留着SOAP式的服务，我工作过程中对接的一些政府如卫生计划委员会、医HIS系统其实依然是保有SOAP的，它活在计算机构建的这一社会的血液里、空气里。

2. 什么是REST?　http://www.fhadmin.org

需要注意的是REST并不是一个标准或者协议，而是一种设计风格，或者说是一个设计web服务的最佳践，其要点如下：

1. 面向资源的URI设计，如user/register;
2. 对资源的操作包括增、删、改、查（和数据库层的操作极为相似）；
3. 连接具有无状态性，即每一次的响应只依赖于这一次的请求；
4. 利用HTTP协议实现以上的设计思想。

非RESTful的设计示意图如下：

image

RESTful的设计示意图如下：

image

3. REST设计

REST的设计利用了HTTP协议的请求option，如GET、POST、PUT、DELETE。设计的简单示意图如：

REST设计

我工作过程中的一些最佳实践是：

1. 对option的选择不应过多，不应死板教条，常用的有GET、POST即可；
2. URI的设计应已名词为主、动词为辅，层次清晰；
3. 参数的设计应已单词为主，少用多个词的驼峰连接形式；
4. 功能与URI或者参数设计冲突时，应以功能实现为主。

4. REST的劣势

a. 一千个读者，一千个哈姆雷特，在设计评审粗糙的情况下，面向资源的URI设计五花八门；

b. URI泛滥，版本管理困难；

c. HTTP option使用不当；

d. REST API 参数、返回值设计不当；