# REST软件架构风格

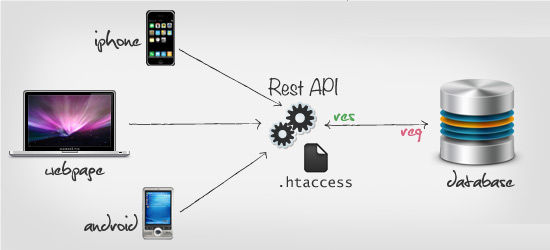
REST即表述性状态传递（英文：Representational State Transfer，简称REST）是Roy Fielding博士在2000年他的博士论文中提出来的一种[软件架构](https://baike.baidu.com/item/%E8%BD%AF%E4%BB%B6%E6%9E%B6%E6%9E%84)风格。它是一种针对[网络应用](https://baike.baidu.com/item/%E7%BD%91%E7%BB%9C%E5%BA%94%E7%94%A8)的设计和开发方式，可以降低开发的复杂性，提高系统的可伸缩性。

资源在网络中以某种表现形式进行状态转移

GET 用来获取资源，  
POST 用来新建资源（也可以用于更新资源），  
PUT 用来更新资源，  
DELETE 用来删除资源。

实用的是如何正确地理解 RESTful架构和设计好RESTful API。

大家都知道"古代"网页都是前端后端融在一起的，比如之前的PHP，JSP等。在之前的桌面时代问题不大，但是近年来移动互联网的发展，各种类型的Client层出不穷，RESTful可以通过一套统一的接口为 Web，iOS和Android提供服务。另外对于广大平台来说，比如Facebook platform，微博开放平台，微信公共平台等，它们不需要有显式的前端，只需要一套提供服务的接口，于是RESTful更是它们最好的选择。在RESTful架构下：



1. URL root:

[https://example.org/api/v1/](https://link.zhihu.com/?target=https%3A//example.org/api/v1/)\*  
[https://api.example.com/v1/](https://link.zhihu.com/?target=https%3A//api.example.com/v1/)\*

2. API versioning:  
可以放在URL里面，也可以用HTTP的header：

/api/v1/

3. URI使用名词而不是动词，且推荐用复数。  
BAD

* /getGirls
* /listOrders
* /retrieveClientByOrder?orderId=1

GOOD

* GET /girls : will return the list of all girls
* POST / girls: will add a girl to the collection
* GET / girls/14 : will retrieve girl #14
* PATCH/PUT / girls/14 : will update girl #14

4. 保证 HEAD 和 GET 方法是安全的，不会对资源状态有所改变（污染）。比如严格杜绝如下情况：

GET /deleteGirls?id=1

5. 资源的地址推荐用嵌套结构。比如：

GET /girls/14/ friends  
UPDATE /profile/primaryAddress/city

6. 警惕返回结果的大小。如果过大，及时进行分页（pagination）或者加入限制（limit）。HTTP协议支持分页（Pagination）操作，在Header中使用 Link 即可。  
7. 使用正确的HTTP Status Code表示访问状态：[HTTP/1.1: Status Code Definitions](https://link.zhihu.com/?target=http%3A//www.w3.org/Protocols/rfc2616/rfc2616-sec10.html)  
8. 在返回结果用明确易懂的文本（String。注意返回的错误是要给人看的，避免用 1001 这种错误信息），而且适当地加入注释。  
9. 关于安全：自己的接口就用https，加上一个key做一次hash放在最后即可。考虑到国情，HTTPS在无线网络里不稳定，可以使用Application Level的加密手段把整个HTTP的payload加密。有兴趣的朋友可以用手机连上电脑的共享Wi-Fi，然后用Charles监听微信的网络请求（发照片或者刷朋友圈）。  
如果是平台的API，可以用成熟但是复杂的OAuth2，新浪微博这篇：[授权机制说明](https://link.zhihu.com/?target=http%3A//open.weibo.com/wiki/%25E6%258E%2588%25E6%259D%2583%25E6%259C%25BA%25E5%2588%25B6%25E8%25AF%25B4%25E6%2598%258E)

**URL定位资源，用HTTP动词（GET,POST,DELETE,DETC）描述操作。**