

当前位置: 译文列表 » Web/WAP应用开发 , 投递原文

RESTful API 设计指南——最佳实践 【已翻译100%】

英文原文: RESTful API Designing guidelines—The best practices
标签: <无>

oschina 推荐于 3周前 (共 11 段, 翻译完成于 03-03) 评论 22

参与翻译 (6人) : ismdeep, Tocy, 无若, butta, 昌伟兄, osbertngok

分享 收藏

仅中文 | 中英文对照 | 仅

Facebook、谷歌、Github、Netflix 和几个其他的科技巨头已经允许开发者和其产品通过 API 调用他们的数据，并为他们提供平台。即使你不是写 API 的专业人士，拥有精美的 API 也对你的应用程序有好处。

关于设计 API 的最好方法，网络上有较长一段时间的争论，但官方也没有对此给出解释。

API 是一个接口，通过它许多开发者可与数据交互。设计优良的 API 使用起来很方便，并给开发者的工作带来便利。API 是开发者的 GUI，如果它设计不合理，开发者会将它替换。所以，开发者的经验是衡量 API 质量的最重要的指标。

API就像一个在舞台上表演的艺术家，其用户就是其观众

Tocy
翻译于 3
2人顶

顶 翻译得不错哦！

1) 术语

以下是与 REST API 相关的重要术语：

- 资源（Resource） 是一个对象或对某物的表示。它有一些相关联的数据，并有一组方法进行操作。例如：动物，学校和员工是资源。这些资源都有着删除，添加，更新操作。
- 集合（Collection） 是一系列资源，例如：公司集合是很多公司的集合。
- URL（统一资源定位符） 是一种路径，可以通过它定位资源并且也可以对它执行一些动作。

ismdee
翻译于 3
1人顶

顶 翻译得不错哦！

2) API 端点（路径）

为了更好地理解，我们给公司写 API，这些公司都有一些员工。/getAllEmployees 是对员工列表进行回应的 API。公司其他 API 大致如下：

- /addNewEmployee
- /updateEmployee
- /deleteEmployee
- /deleteAllEmployees
- /promoteEmployee
- /promoteAllEmployees

并且将有大量的和这些操作不同的 API 端点，它们包含大量冗余的行为。因此，当 API 数量增加时，这些 API 端点将很难维护。

哪里不对？

每个 URL 代表一种资源（Resource），所以 URL 中只能有名词，不能有动词。API 路径 /addNewEmployee 包含了操作 addNew 和资源名称 Employee。

ismdee
翻译于 3
1人顶

顶 翻译得不错哦！



那么怎样算是正确的方式？

/companies 是一个很好的不包含操作的例子。但是问题来了，我们该怎样告诉服务器我们要进行的操作呢？新增，删除，还是更新？

这时 HTTP 方法（GET，POST，DELETE，PUT）（也称为动词）就可以起到作用了。

资源在 API 端点中应该总是复数，如果我们想访问资源的一个实例，我们可以传递 URL 中的 id。

- 方法 GET 路径 /companies 是获取所有公司的列表。
- 方法 GET 路径 /companies/34 是获取公司34的详细信息。
- 方法 DELETE 路径 /companies/34 是删除公司34。



ismd6

翻译于 3

1人顶

顶 翻译得不错哦！

在其他的一些使用案例中，如果我们有一些资源在某个资源之下，例如，一个公司的员工，那么在这样的例子中 API 的 endpoint（端点） 就应该是这样的：

- GET /companies/3/employees 可以取得编号为3的公司的员工列表
- GET /companies/3/employees/45 可以取得编号为3的公司的45号员工的细节信息
- DELETE /companies/3/employees/45 可以删除编号为3的公司的45号员工
- POST /companies 可以创建一个新公司并返回新创建公司的细节信息

现在这样，API 是不是更严谨和一致了呢？

结论：路径应该包含资源的复数形式，HTTP 方法应该定义成各种行为在资源上执行。



无若

翻译于 3

2人顶

顶 翻译得不错哦！

3) HTTP 方法 (动词)

HTTP 定义了几组方法，这些方法给出了对资源要执行的操作类型。

URL 是一个句子，其中资源是名词，HTTP 方法是动词。

主要的HTTP方法如下：

1. GET 方法从资源请求数据，不产生多余结果。
例如： /companies/3/employees 会返回公司3的所有雇员列表。
2. POST 方法请求服务器在数据库中创建资源，这主要用于提交 Web 表单时。
例如： /companies/3/employees 创建一个公司3的新雇员。
POST 是非幂等的，这意味着多个请求将会有不同的效果。
3. PUT 方法请求服务器更新资源或创建资源（如果不存在的话）。
例如： /companies/3/employees/john 将请求服务器在公司3的雇员集合中更新或在不存在的情况下创建关于 john 的资源。
PUT 是幂等的，这意味着多次请求具有相同的效果。
4. DELETE 方法将请求的资源或实例从数据库中删除。
例如： /companies/3/employees/john/ 将请求服务器从公司3的雇员集中删除 john 资源。

HTTP 中还有很多其他方法，我们将在另一篇文章中讨论。



Tocy

翻译于 3

1人顶

顶 翻译得不错哦！

4) HTTP 响应状态码

当客户端通过 API 向服务器发起请求时，无论请求是失败的、通过的还是错误的，客户端应该获得反馈。HTTP 状态码是一堆标准化的数值码，在不同的情况下具有不同的解释。服务器应始终返回正确的状态码。



Tocy

翻译于 3

1人顶

以下是 HTTP 状态码的主要分类：

顶 翻译得不错哦！

2xx (成功类别)

这些状态代码表示请求的操作已被服务器接收到并成功处理。

- 200 Ok：标准的 HTTP 响应，表示 GET、PUT 或 POST 的处理成功。
- 201 Created：在创建新实例时，应返回此状态代码。例如，使用 POST 方法创建一个新的实例，应该始终返回 201 状态码。
- 204 内容不存在：表示请求已被成功处理，但并未返回任何内容。

DELETE算是其中一个很好的例子。

API DELETE /companies/43/employees/2 将删除员工 2，作为响应，我们不需要在该 API 的响应正文中的任何数据，因为我们明确地要求系统将其删除。如果有任何错误发生，例如，如果员工 2 在数据库中不存在，那么响应码将不是 2xx 对应的成功类别，而是 4xx 客户端错误类别。

3xx (重定向类别)

- 304 未修改：表示客户端的响应已经在其缓存中。因此，不需要再次传送相同的数据。

4xx (客户端错误类别)

这些状态代码表示客户端发起了错误的请求。

- 400 错误请求：表示客户端的请求没有被处理，因为服务器不能理解客户端请求的是什么。
- 401 未授权：表示客户端不被允许访问该资源，需要使用指定凭证重新请求。
- 403 禁止访问：表示请求是有效的并且客户端已通过身份验证，但客户端不被允许以任何理由访问对应页面或资源。例如有时授权的客户端不被允许访问服务器上的目录。
- 404 未找到：表示所请求的资源现在不可用。
- 410 资源不可用：表示所请求的资源后续不再可用，该资源已被移动。



Tocy
翻译于 3
3人顶

顶 翻译得不错哦！

5xx (服务器错误类别)

- 500是服务器内部错误，表示请求已经被接收到了，但服务器被要求处理某些未预设的请求而完全混乱。
- 503服务不可用表示服务器已关闭或无法接收和处理请求。大多数情况是服务器正在进行维护。

5) 字段套管约定

您可以遵循任何套管约定，但要确保其在应用程序中一致。如果请求主体或响应类型是JSON，那么请按照camelCase(驼峰命名法)来保持一致。



ismde
翻译于 3
1人顶

顶 翻译得不错哦！

6) 搜索、排序、过滤和分页

这些行为都只是针对一个数据集进行的查询。由于还没有一套新的 API 来处理这些行为。因此，我们需要向 GET 方法的 API 附加查询参数。

我们来理解几个例子看它们是如何实现这些行为的。

- 排序(sorting)，客户端想要获得排序后的公司列表，GET /companies 端点应当接受多个查询排序参数。
例如，GET /companies?sort=rank_asc 将根据等级以升序的方式对公司进行排序。
- 过滤(Filtering)，用来过滤数据集，我们可以通过查询参数传递不同的选项。
例如，GET /companies?category=banking&location=india 将根据公司类别为银行以及所处位置为印度来过滤公司的列表数据。



昌伟5
翻译于 3
2人顶

顶 翻译得不错哦！



1

本文中的所有译文仅用于学习和交流目的，转载请务必注明文章译者、出处、和本文链接

我们的翻译工作遵照 [CC 协议](#)，如果我们的工作有侵犯到您的权益，请及时联系我们

评论(22)

😊 📄

Ctrl/CMD+Enter 发表评论

落花追魂 发表于 2017-03-03 14:58
看看能不能发链接 <http://docdev.cn>

落花追魂 发表于 2017-03-03 14:58
很明显 不能发

橙子先生 发表于 2017-03-04 22:11
请问响应数据的规范格式是什么样的？

65535 发表于 2017-03-06 09:40
国内的话 http状态码 保留意见。

grath 发表于 2017-03-06 10:32
稳定的api标注版本号

grath 发表于 2017-03-06 10:33
引用来自“橙子先生”的评论
请问响应数据的规范格式是什么样的？

我们是 {code: 200, data: {boom: "hello"}, message: "提示信息", others:}
有些特别的数据直接丢到了 others里面，也可能是不适合放到data字段的

橙子先生 来自 [Android](#) 发表于 2017-03-08 08:27
没有安全性方面的，比如说签名这块

橙子先生 来自 [Android](#) 发表于 2017-03-08 08:27
没有安全性方面的，比如说签名这块



hantsy 发表于 2017-03-08 09:22



引用来自“橙子先生”的评论
请问响应数据的规范格式是什么样的？

引用来自“grath”的评论
我们是 {code: 200, data: {boom: “hello”}, message: “提示信息”, others:}
有些特别的数据直接丢到了 others里面, 也可能是不适合放到data字段的

这种太扯了, 完全没必要。

直接使用 HAL 就行了, 错误信息有标准的 Problem API 和 Error 去描述返回。。。一般2XX 状态都是正确处理结果, 直接返回结果就行了。



hantsy 发表于 2017-03-08 09:26



REST API 设计只要看 Github 和 Heroku API 设计就足够了。这篇文章远远不够指导 API 设计, Richardson Mature Model 为规范 REST 提出了三个级别的 Level, 国内的大部分公司处于 Level 0, 也就是根本就不是 REST API (包括某些大厂)。



voidint0 发表于 2017-03-08 09:28



引用来自“橙子先生”的评论
请问响应数据的规范格式是什么样的？

引用来自“grath”的评论
我们是 {code: 200, data: {boom: “hello”}, message: “提示信息”, others:}
有些特别的数据直接丢到了 others里面, 也可能是不适合放到data字段的

引用来自“hantsy”的评论
这种太扯了, 完全没必要。

直接使用 HAL 就行了, 错误信息有标准的 Problem API 和 Error 去描述返回。。。一般2XX 状态都是正确处理结果, 直接返回结果就行了。

HAL是什么？



waylau 发表于 2017-03-08 09:34



补充下最佳实践：<https://waylau.com/best-practices-for-better-restful-api/>



月之舞步 发表于 2017-03-08 10:18



如果在 GET 方法中附加了很多查询参数, 会造成 URI 太长, 服务器可能会响应 414 的 HTTP 状态, 表示这个 URI 太长, 在这种情况下, 我们也可以将参数传递给 POST 方法的请求体中。
这里我就有疑问了, 我这参数很多, 需要用POST去查询的时候, 岂不是变成了创建了? 或者说, 我在设计API的时候, 就要考虑这种冲突, 防止这种冲突出现



卓尔 发表于 2017-03-08 10:22



这篇文章写的有深度



_Untitled 发表于 2017-03-08 10:34



引用来自“橙子先生”的评论
请问响应数据的规范格式是什么样的？

引用来自“grath”的评论
我们是 {code: 200, data: {boom: “hello”}, message: “提示信息”, others:}
有些特别的数据直接丢到了 others里面, 也可能是不适合放到data字段的

引用来自“hantsy”的评论
这种太扯了, 完全没必要。

直接使用 HAL 就行了, 错误信息有标准的 Problem API 和 Error 去描述返回。。。一般2XX 状态都是正确处理结果, 直接返回结果就行了。

引用来自“voidint0”的评论
HAL是什么？

同问。还有究竟数据应该怎么返回, 能否给出示例。



天涯1979 来自 iPhone 发表于 2017-03-08 11:37



引用来自“橙子先生”的评论
没有安全性方面的, 比如说签名这块



@橙子先生 对，缺少基本的认证等

- 

橙子先生 来自 [Android](#) 发表于 2017-03-08 12:20

引用来自“橙子先生”的评论

没有安全性方面的，比如说签名这块

引用来自“天涯1979”的评论

@橙子先生 对，缺少基本的认证等

这个怎么能叫最佳实践呢？基本上都是入门级别
- 

QiHaiYan 发表于 2017-03-08 13:17

引用来自“橙子先生”的评论

没有安全性方面的，比如说签名这块

Secure REST API with oauth2
<https://my.oschina.net/hiease/blog/742158>
- 

CheckStyle 发表于 2017-03-08 14:09

无招胜有招，不要过于拘泥这些
- 

not_empty 发表于 2017-03-08 17:33

这种url 在后端做权限的时候不好弄