

Http 参数格式约定, 接受反驳

作者: [crick77](#)

原文链接: <https://ld246.com/article/1530259248612>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

前提

SpringCloud 是 http 协议传输，接口协议自由且松散，所以需要强有力的约定来限制接口。

服务内部实现，可以基于个人习惯，但是必须提供风格一致的输入输出，减少开发与维护成本。

疑问

Request 疑问

接口不兼容升级，如何区分版本？

如何使用 request method? include <code>GET</code>、<code>PUT</code>、<code>POST</code>、<code>DELETE</code> and so on.

入参如何定义? include <code>path</code>、<code>param</code>、<code>Request ody</code>、<code>Header</code>

Response 疑问

常见格式如下:

```
<code class="language-json highlight-chroma"><span class="highlight-line"><span class="highlight-cl"><span class="highlight-p">{</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nt">"code"</span><span class="highlight-p">:</span><span class="highlight-mi">0</span><span class="highlight-p">,</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nt">"data"</span><span class="highlight-p">:</span><span class="highlight-p"></span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nt">"name"</span><span class="highlight-p">:</span><span class="highlight-s">"zhangsan"</span><span class="highlight-p">,</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nt">"age"</span><span class="highlight-p">:</span><span class="highlight-mi">4</span></span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nt">"message"</span><span class="highlight-p">:</span><span class="highlight-2">"ok"</span><span class="highlight-p">,</span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-nt">"isRetry"</span><span class="highlight-p">:</span><span class="highlight-kc">false</span></span></span></span><span class="highlight-line"><span class="highlight-cl"><span class="highlight-p">}</span></span></code></pre>
```


code 如何定义? code=0 表示成功 or 返回 http_code 值?

error_code 如何定义? 扩展问题为，前端如何根据 code 进行提示渲染?

预期

Request 规范

接口无法兼容时，升级版本。非高频事件。默认接口无版本号，升级时 path 中增加版本，从/v2 开始。

针对 request method，不完全依赖 restful 定义，仅使用 GET POST 两种方法。查询接口使用 GET，业务逻辑接口使用 POST。

入参定义

- 不使用 `@PathVariable`, 难做到单参数满足业务需求, 不方便扩展。
- 使用 `@RequestParam` 描述参数, 不符合时返回 `400` 状态。
- 表单提交使用 `POST`, 通过 `application/json` 利用 `@RequestBody` 转换 javaBean。
- `Header` 传输通用信息, 如 ip、deviceId、token 等, 且在 nginx 转发时, 同转发 header 参数。

 - 通过 `http_status` 返回 `error_code`。常用如:400、500 及自定义频段。此时 response 内容为 `error_message`。 - 前端基于 `http_status` 查询对应渲染文案 (含国际化)。 <p>不使用 json 定义 code 的理由如下: </p> - 更符合 http 规范 - 减少性能损耗。读取 `http_status` 为 error 时, 不需要解析 response 内容。