

WEB架构 服务化

2014 / @oneoo

UPYUN | PHP

云存储对外API接口服务

集群规模 8 台 8G 内存 4 核服务器，5000/qps

WEB 开发中的常见业务

- | | | |
|--------|----|--------|
| • 数据库 | 网络 | |
| • 缓存 | 网络 | |
| • 模板 | 本地 | CPU密集型 |
| • 文件存储 | 本地 | IO密集型 |
| • 其他 | 本地 | CPU密集型 |
| ▪ 图片处理 | | |
| ▪ 音频处理 | | |
| ▪ 视频处理 | | |
| ▪ ... | | |

我们都想快

但任务繁多，无法快起来。只有轻装上阵



让我们把‘包袱’都扔了

- 文件存储
- 图片处理
- 音频处理
- 视频处理
- 甚至复杂的数据库操作
-

这些统统不要！

但事情还得做的
换个形式，服务化

把这些大头给抽象出来

所以我们有

- 文件云存储服务（150+台20TB存储服务器）
- 图片处理服务（16台32核服务器）
- 音视频处理服务（8台32核服务器）
- MySQL、Redis等（10台32G内存服务器）
- 。。。

有接近200台服务器提供联合服务
均为HTTP REST接口的集群模式
每个服务自主负载均衡和容灾

甚至核心API服务（账号开设、创建空间等）

WEB项目中只剩下业务逻辑!



没事一身轻

屁爆了，甩个尾都没问题

虽然处理任务扔出去了，但我们还得捕获结果吧？

只是从原来的文件IO、CPU计
算换成网络操作

重点是网络IO可非堵塞 :)

那就是说同样的时间，我们
可以做更多的事儿！

异步并发

```
Main Job {  
    r1 = insert database {...}  
    r2 = save file {...}  
    r3 = process image {...}  
    wait(r1,r2,r3);  
    render template;  
    return;  
}
```

如果是PHP，可以引入Yar 或 Swoole

假设一个业务，分数据库、写文件、处理缩略图3个部分，串行操作需耗时300ms


现在把这3个处理任务，分派到3个集群

有4台服务器并行处理，耗时100MS

额外收获：轻耦合架构

多个子服务可横行扩容、可降级、灰度发布

更容易定位故障

A full-page photograph of a male golfer on a green golf course under a clear blue sky. The golfer is wearing a bright green short-sleeved polo shirt, brown and red plaid trousers, white golf shoes, and a white bucket hat. He is standing on a green, looking upwards with his right hand open and palm facing up. To his left is a yellow flagstick with a white flag, and a golf ball sits on the grass nearby. In the background, there are trees, a sand trap, and a white golf cart parked on the grass. Overlaid on the image are three lines of text in white and black boxes.

原来只是屌丝，累死累活

而现在你是管理者！

有多个得力的“员工”给你卖力了！

但是

需要注意

并行的事务处理很蛋疼

需做好

临时状态记录
回滚标记

...

(Q & A)

@oneoo

<https://UPYUN.com>