

# 并发不是简单的并行

Concurrency is not just Parallelism

简介流水线作业



再理解Rob Pike的

Concurrency is not Parallelism (it's better)

<http://concur.rspace.googlecode.com/hg/talk/concur.html#landing-slide>



# 我是谁

@sunfmin

<http://weibo.com/sunfmin>

<https://github.com/sunfmin>

.....

孙凤民

Felix Sun

就职于 thePlant 

<http://theplant.jp>



# 统筹方法

华罗庚



# 串行 泡茶



洗水壶(1分钟)



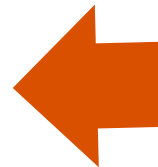
烧开水(15分钟)



洗茶壶(2分钟)



洗茶杯(2分钟)



泡茶(5分钟)



拿茶叶(1分钟)



# 总用时 26分钟

洗水壶 (1分)

烧开水 (15分)

洗茶壶 (2分)

拿茶叶 (1分)

泡茶 (5分)

洗茶杯 (2分)



如果我要泡4杯茶？

并行









问题：

每26分钟生产一杯茶



# 并发I



洗水壶(1分钟)



烧开水(15分钟)



洗茶壶(2分钟)



拿茶叶(1分钟)



泡茶(5分钟)



洗茶杯(2分钟)



21分钟



# 烧开水最费时间！ 那么并发他！



烧开水(15分钟)





# 并发I



并发5, 每3分钟烧出一壶开水



$$1 + 3 + 5 = 9 \text{分}$$



# 泡茶（5分钟） 已经成为瓶颈



泡茶(5分钟)



# 并发2



洗水壶(1分钟)



洗茶壶(2分钟)



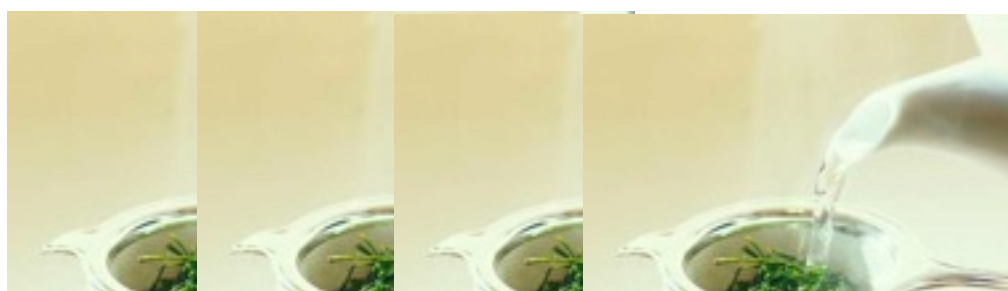
拿茶叶(1分钟)



烧开水( 烧开水( 烧开水(15分钟) 烧开水( 烧开水(15分钟)



并发5, 每3分钟烧出一壶开水



泡茶 泡茶( 泡茶 泡茶(5分钟)



洗茶杯(2分钟)



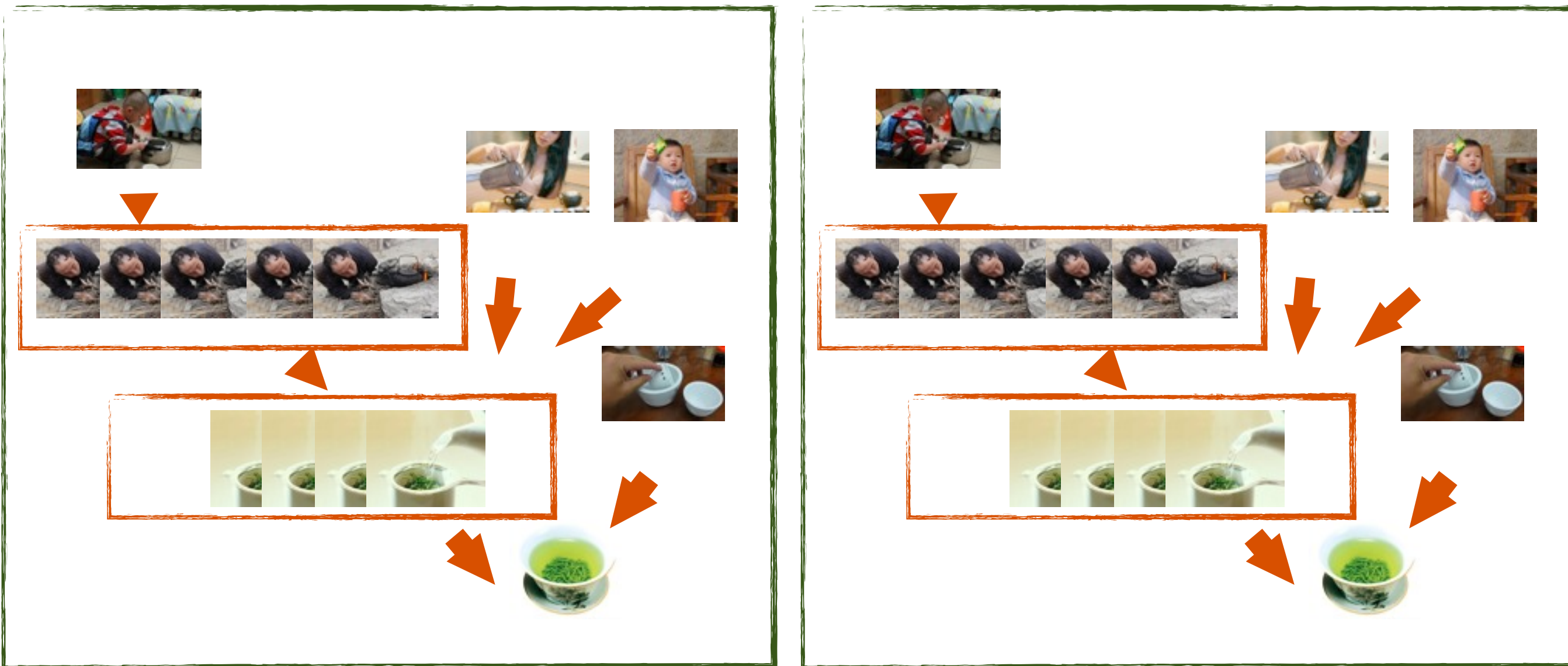
并发4, 每1.25分钟 泡一壶茶

$$1 + 3 + 2 = 6 \text{分}$$





# 并发3



每3分钟一壶





# 并发大于并行， 包含并行



# 并发包含的元素

1. 同时执行的任务: Goroutine
2. 任务之间的协调: Channel



```
1
2 type WaterPot struct {
3     Size int
4 }
5
6 var WashedWaterPots = make(chan WaterPot)
7
8 func BoilWater() {
9     for {
10         pot := <- WashedWaterPots
11         Boil(pot)
12     }
13 }
14
15 go BoilWater()
16
```



# 演示



# 广告贴

ThePlant将要公测企业内部交流平台 Qortex™

完全Go语言开发 <http://qortex.net>

欢迎关注Twitter: @theqortex

