PHPCON
www.phpconchina.com

Swoole 5 带 PHP 走向远方

学而思网校 韩天峰

PHPCon 历年完整 PPT 下载站：

https://github.com/ThinkDevelopers/PHPConChina

自我介绍

1. 好未来集团 学而思网校 首席架构师
2. Swoole 开源项目创始人、核心开发者
3. PHP & Swoole 布道师

新一代的开发者逐渐成为中坚力量，比如 郭新华、Twosee

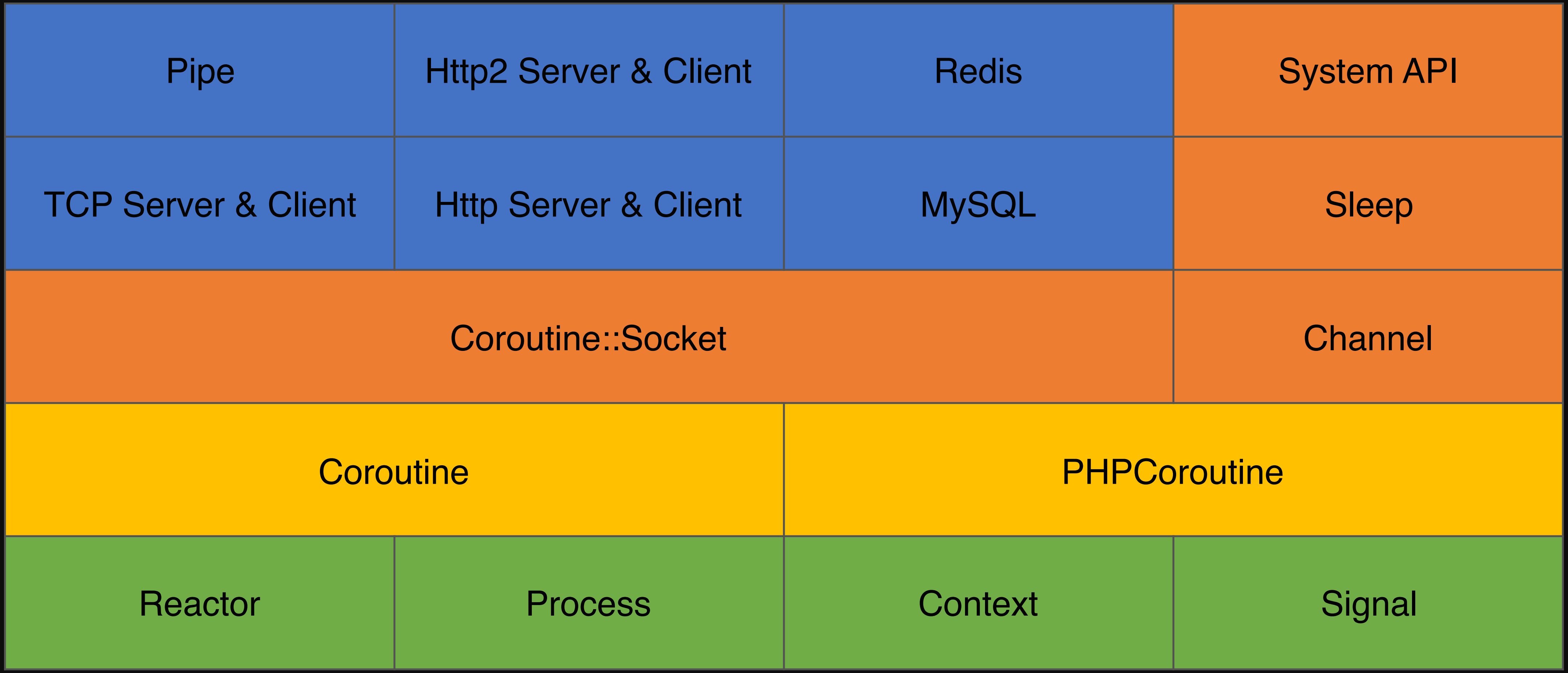
Swoole 4.4

1. Hook： curl、proc_open、shell_exec、gethostbyname

2. 重构：内核协程化

3. 移除限制：可创建无数个协程，嵌套层数不限

全新的代码架构



Pipe

Http2 Server & Client

Redis

System API

TCP Server & Client

Http Server & Client

MySQL

Sleep

Coroutine::Socket

Channel

Coroutine

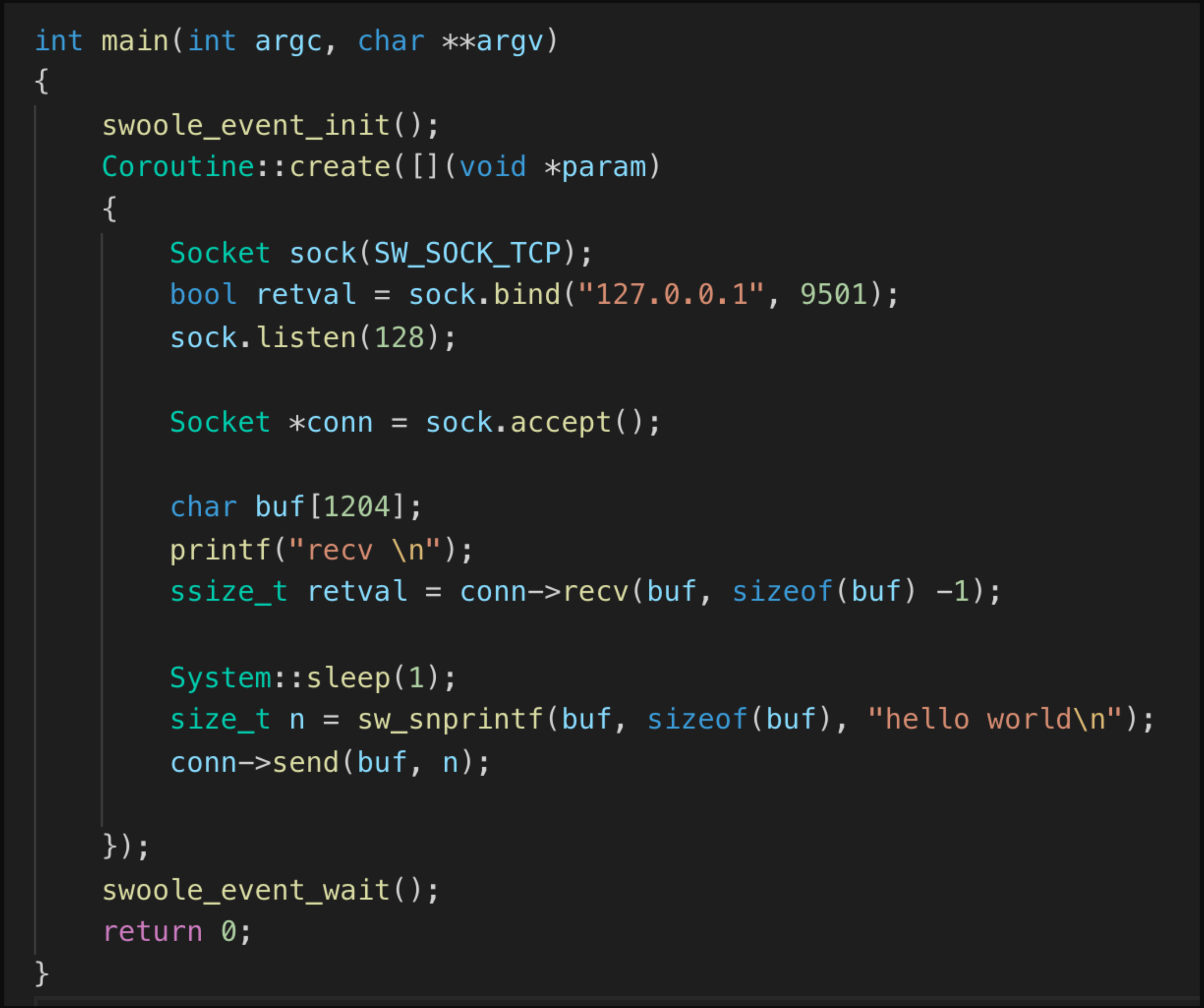
PHPCoroutine

Reactor

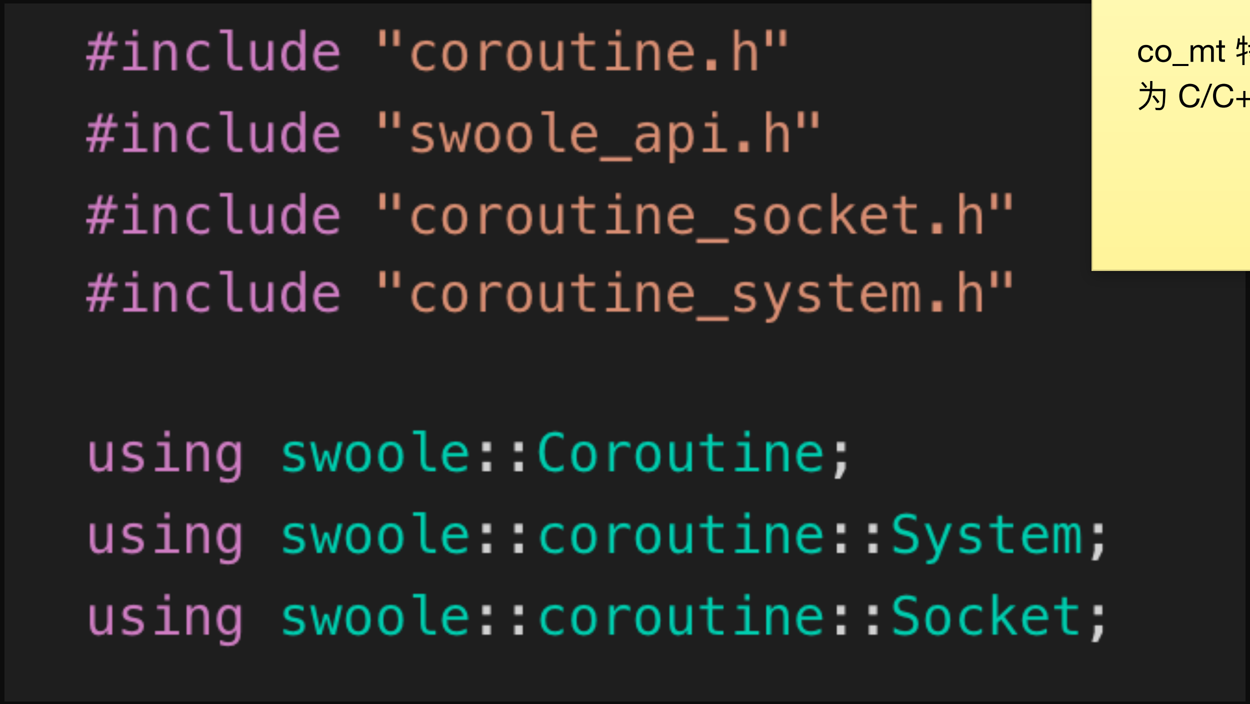
Process

Context

Signal



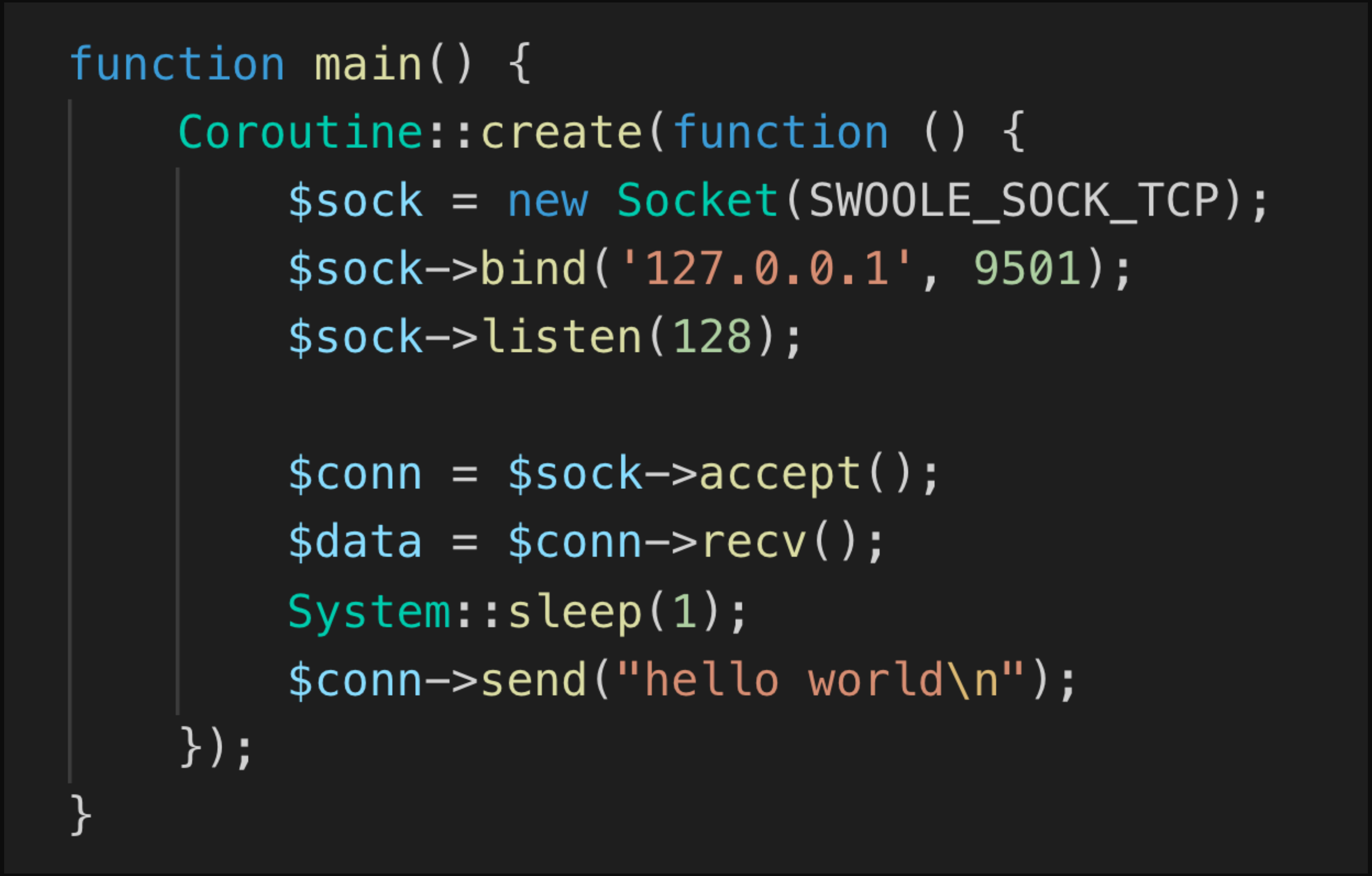
```
int main(int argc, char **argv)
{
    swoole_event_init();
    Coroutine::create([](void *param)
    {
        Socket sock(SW_SOCK_TCP);
        bool retval = sock.bind("127.0.0.1", 9501);
        sock.listen(128);

        Socket *conn = sock.accept();

        char buf[1204];
        printf("recv \n");
        ssize_t retval = conn->recv(buf, sizeof(buf) -1);

        System::sleep(1);
        size_t n = sw_snprintf(buf, sizeof(buf), "hello world\n");
        conn->send(buf, n);

    });
    swoole_event_wait();
    return 0;
}
```



```
#include "coroutine.h"
#include "swoole_api.h"
#include "coroutine_socket.h"
#include "coroutine_system.h"

using swoole::Coroutine;
using swoole::coroutine::System;
using swoole::coroutine::Socket;
```

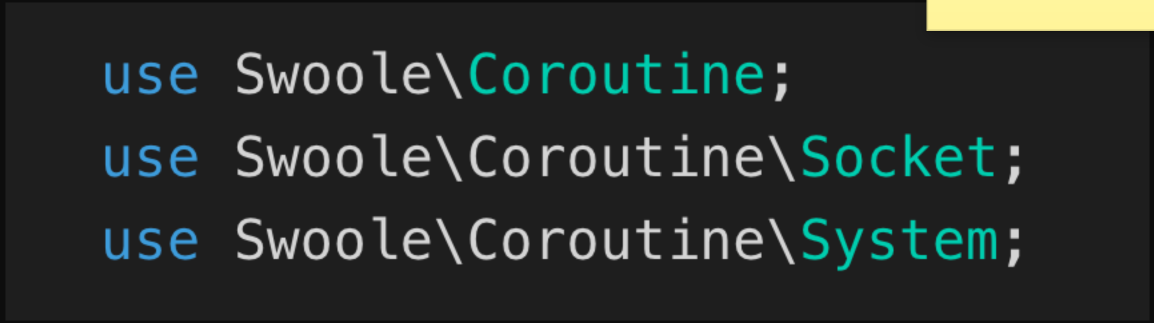
引入头文件
使用命名空间

co_mt 特性，多线程 + 协程，完全可以作为 C/C++ 协程库

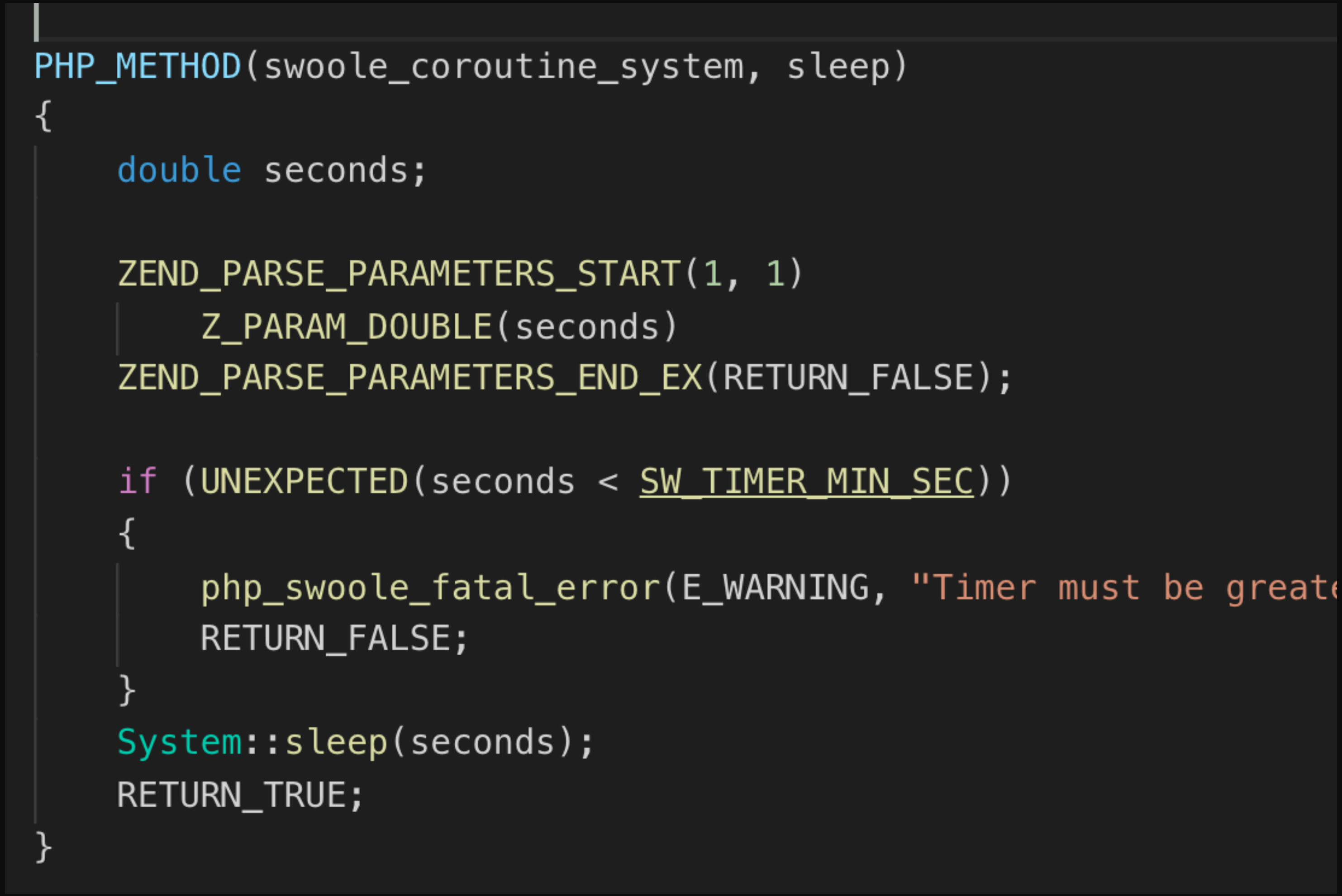


```
function main() {
    Coroutine::create(function () {
        $sock = new Socket(SWOOLE_SOCK_TCP);
        $sock->bind('127.0.0.1', 9501);
        $sock->listen(128);

        $conn = $sock->accept();
        $data = $conn->recv();
        System::sleep(1);
        $conn->send("hello world\n");
    });
}
```

头文件、类 与 C++ 的代码完全一致。



```php
use Swoole\Coroutine;
use Swoole\Coroutine\Socket;
use Swoole\Coroutine\System;
```



```
PHP_METHOD(swoole_coroutine_system, sleep)
{
    double seconds;

    ZEND_PARSE_PARAMETERS_START(1, 1)
        Z_PARAM_DOUBLE(seconds)
    ZEND_PARSE_PARAMETERS_END_EX(RETURN_FALSE);

    if (UNEXPECTED(seconds < SW_TIMER_MIN_SEC))
    {
        php_swoole_fatal_error(E_WARNING, "Timer must be greate
        RETURN_FALSE;
    }
    System::sleep(seconds);
    RETURN_TRUE;
}
```

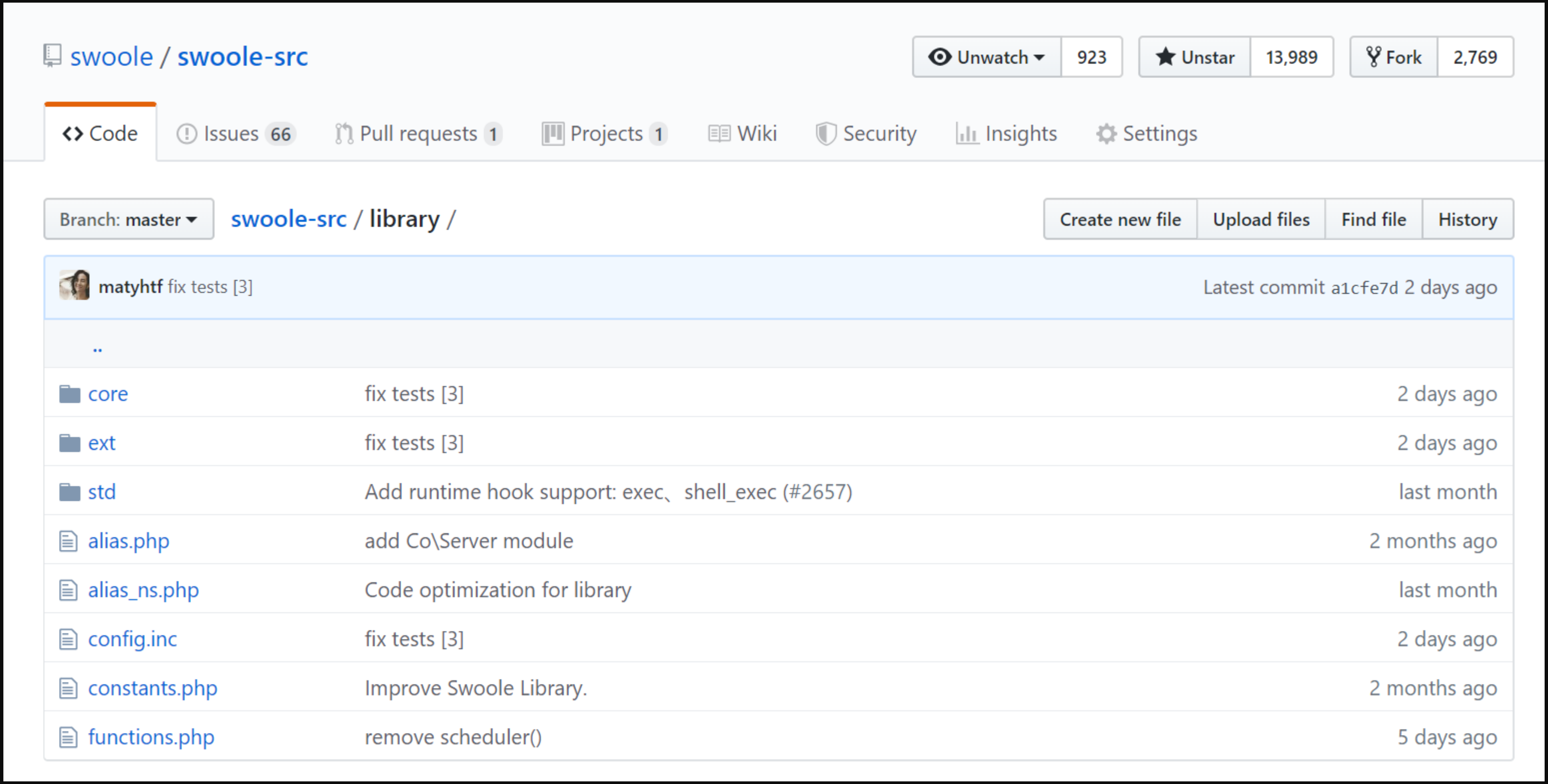
底层与应用层高度统一：内核协程化

Swoole PHP Library

1. 用 PHP 代码来实现 Swoole 模块

2. 无需 include/require，无需 composer

3. 安装即可使用



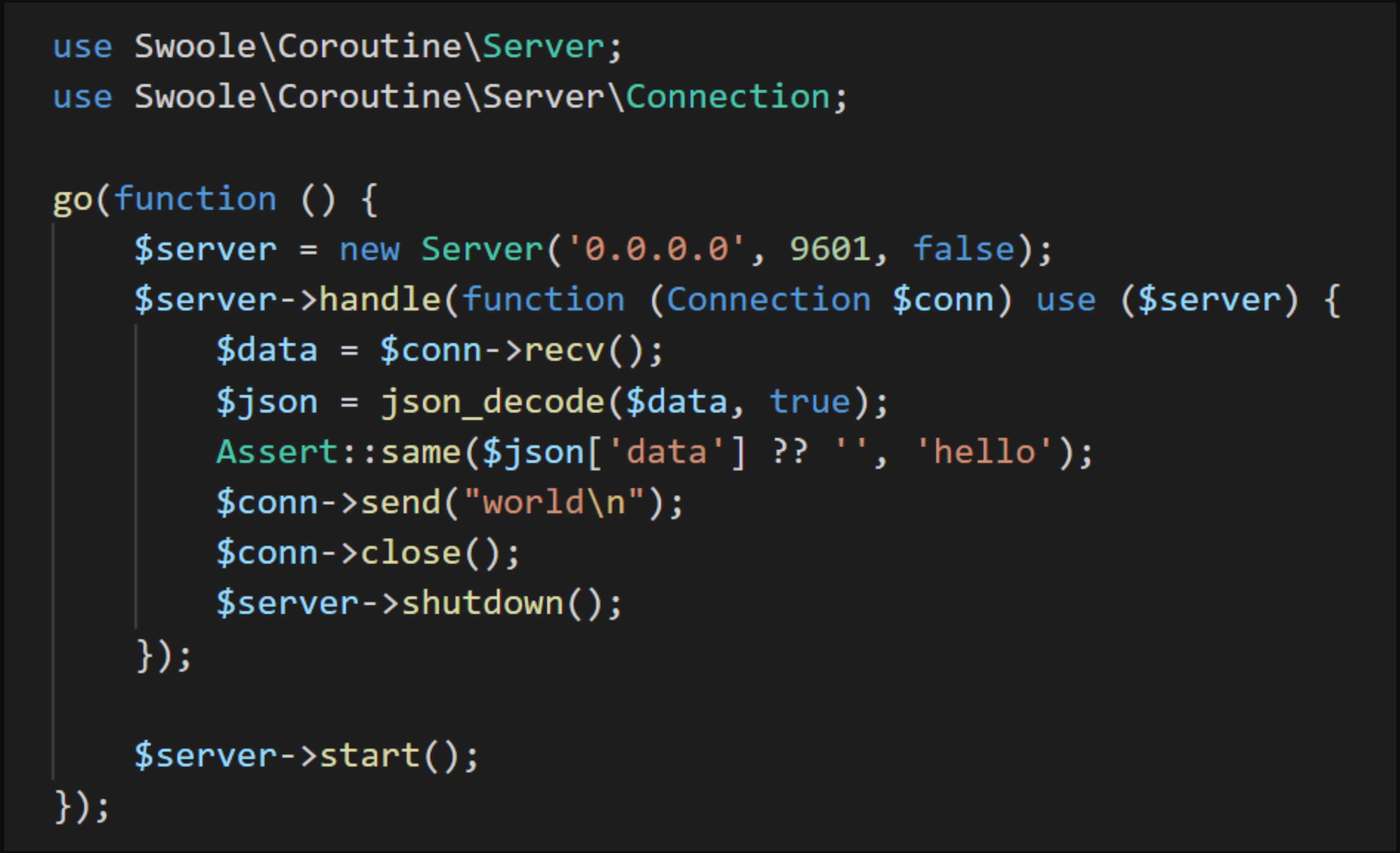
swoole / swoole-src

swoole-src / library /

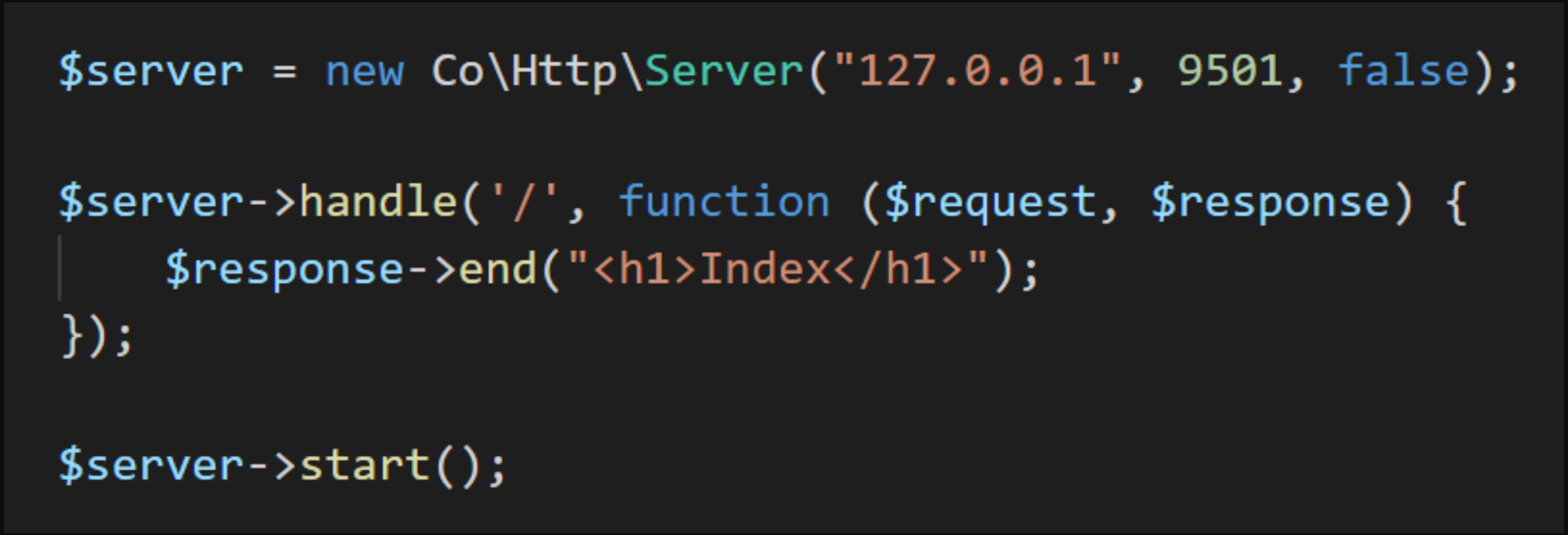
matyhtf fix tests [3] — Latest commit `a1cfe7d` 2 days ago

..		
core	fix tests [3]	2 days ago
ext	fix tests [3]	2 days ago
std	Add runtime hook support: exec、shell_exec (#2657)	last month
alias.php	add Co\Server module	2 months ago
alias_ns.php	Code optimization for library	last month
config.inc	fix tests [3]	2 days ago
constants.php	Improve Swoole Library.	2 months ago
functions.php	remove scheduler()	5 days ago

协程化 Server 、HttpServer



```php
use Swoole\Coroutine\Server;
use Swoole\Coroutine\Server\Connection;

go(function () {
    $server = new Server('0.0.0.0', 9601, false);
    $server->handle(function (Connection $conn) use ($server) {
        $data = $conn->recv();
        $json = json_decode($data, true);
        Assert::same($json['data'] ?? '', 'hello');
        $conn->send("world\n");
        $conn->close();
        $server->shutdown();
    });

    $server->start();
});
```



```php
$server = new Co\Http\Server("127.0.0.1", 9501, false);

$server->handle('/', function ($request, $response) {
    $response->end("<h1>Index</h1>");
});

$server->start();
```

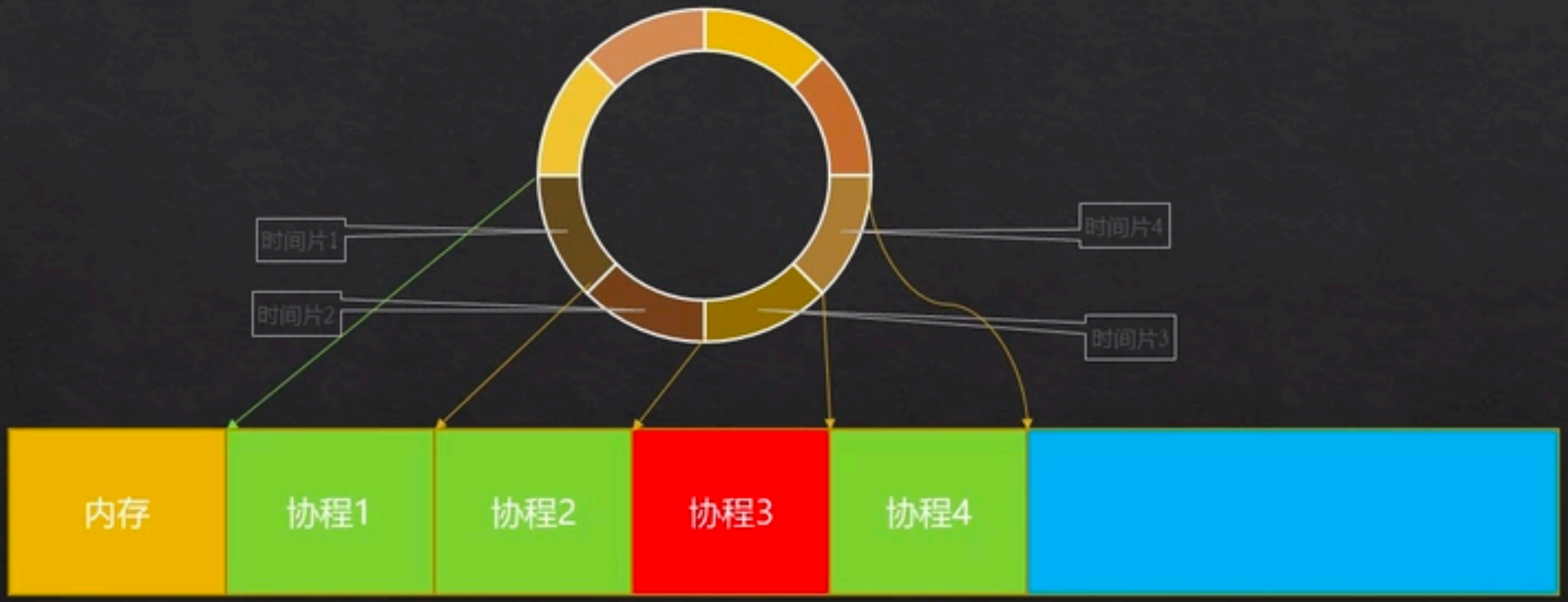
中断调度器

1. 协程可以像线程一样，抢占 CPU 时间，实现了多线程的效果

2. 使用 中断线程 + VM Interrupts 机制实现

3. 相当于 Erlang 的能力，C++/Golang 静态语言无法实现

CPU时间片

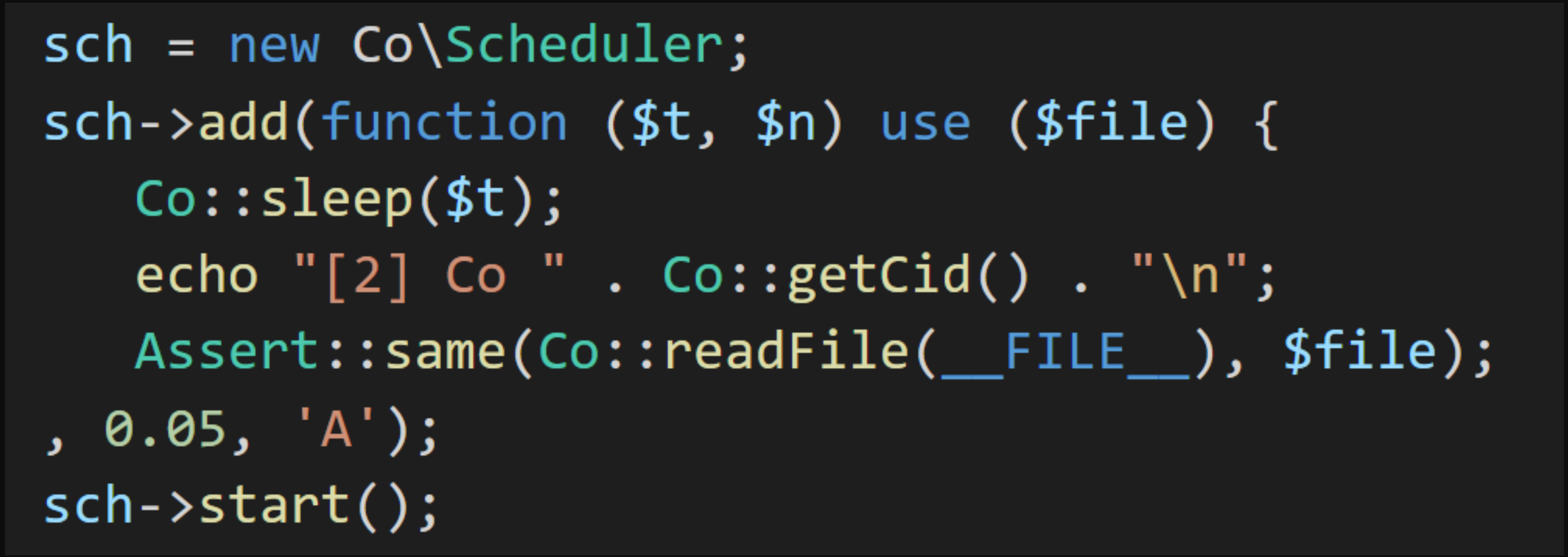


操作全局变量是否需要加锁?

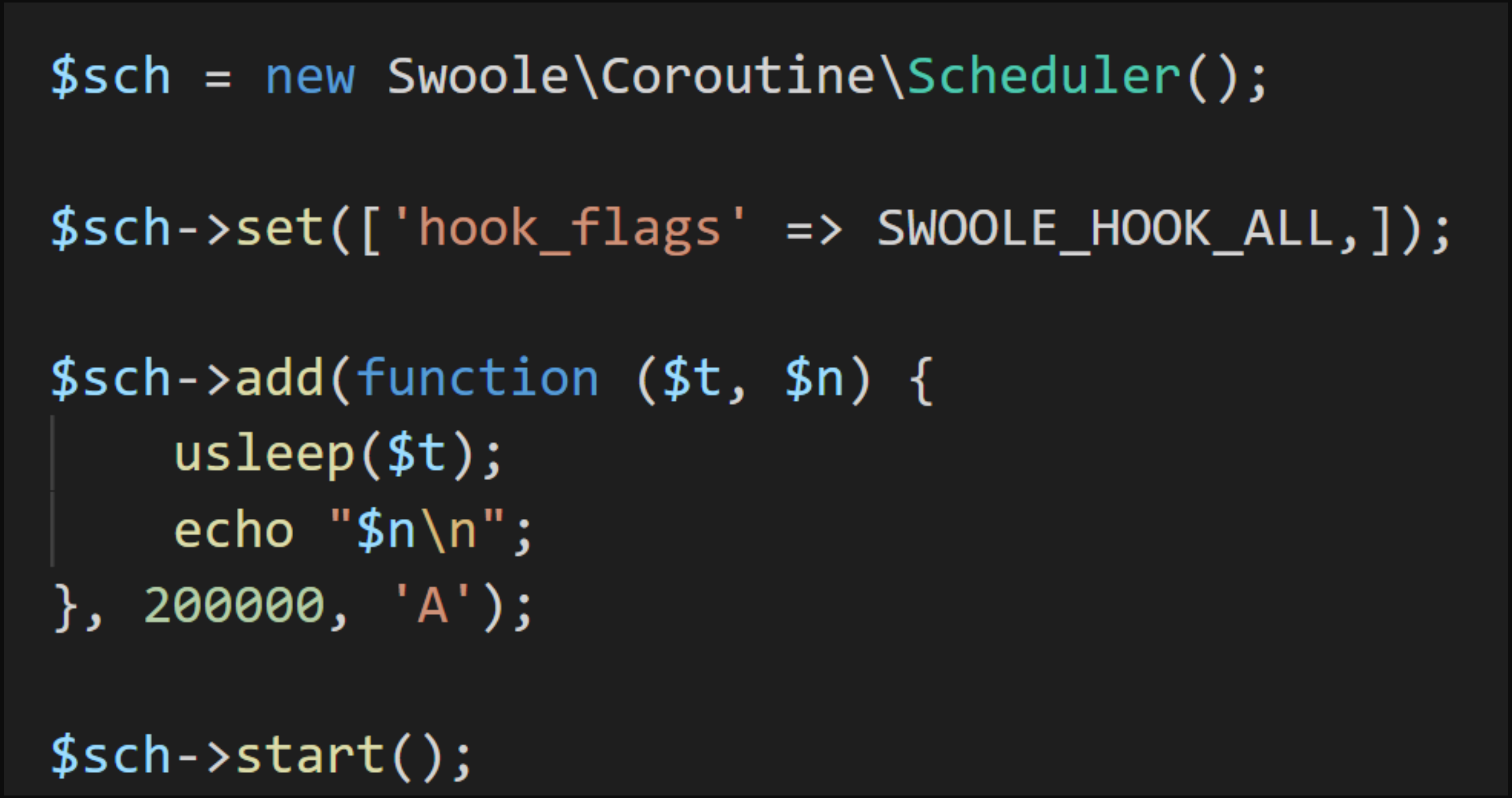
1. 临时关闭中断调度
2. Coroutine::disableSchedule()
3. Coroutine::enableSchedule()

历史包袱

1. 移除异步回调式 API
2. 移除杂项模块，如 Buffer、Serialize 等，更专注
3. 程序入口统一化，移除 Event::wait
4. Runtime::enableCoroutine 改为 hook_flags 配置



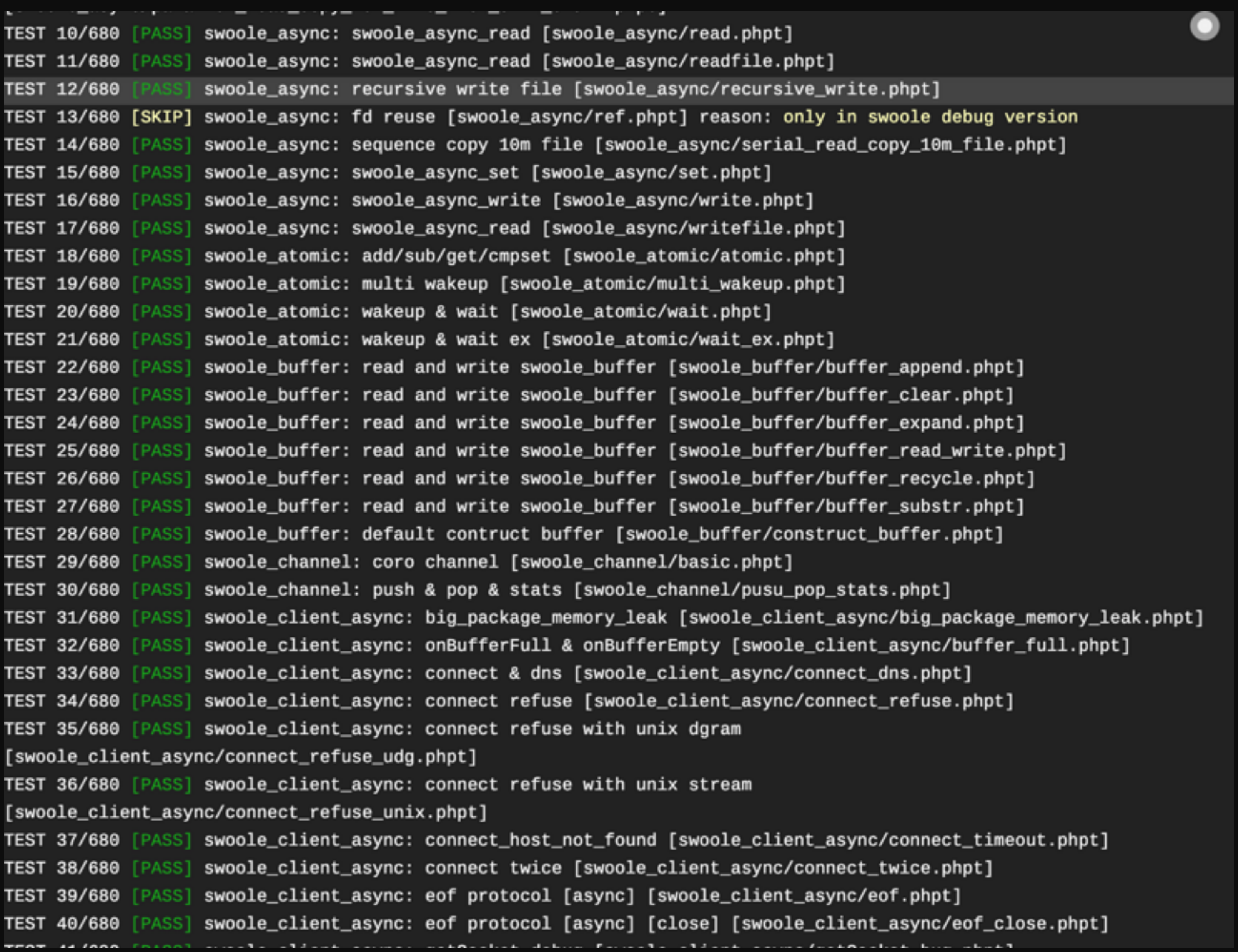
```php
sch = new Co\Scheduler;
sch->add(function ($t, $n) use ($file) {
    Co::sleep($t);
    echo "[2] Co " . Co::getCid() . "\n";
    Assert::same(Co::readFile(__FILE__), $file);
, 0.05, 'A');
sch->start();
```



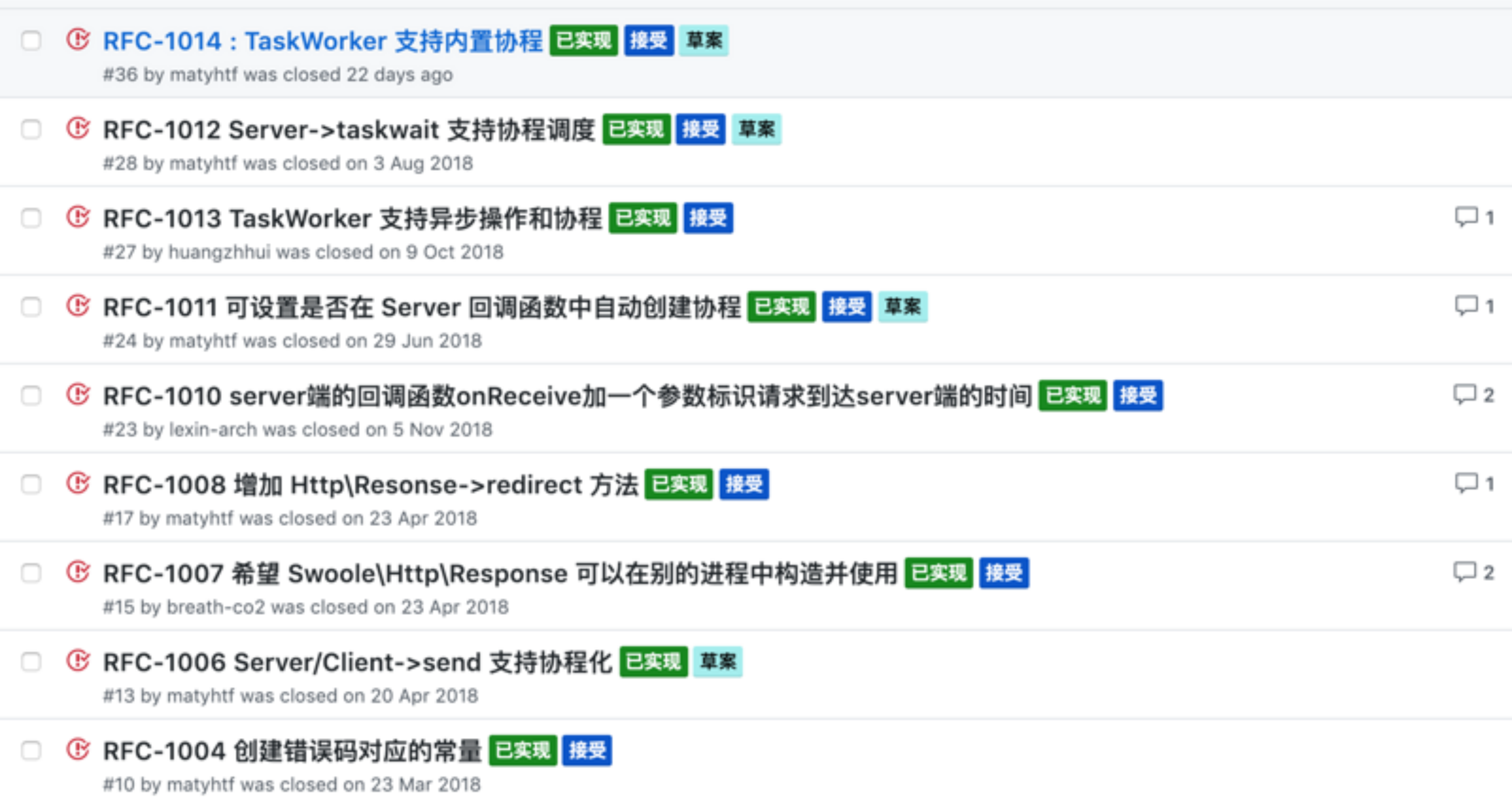
```php
$sch = new Swoole\Coroutine\Scheduler();

$sch->set(['hook_flags' => SWOOLE_HOOK_ALL,]);

$sch->add(function ($t, $n) {
    usleep($t);
    echo "$n\n";
}, 200000, 'A');

$sch->start();
```

研发管理

- 单元测试
- CI
- RFC
- Code Review



```
TEST 10/680 [PASS] swoole_async: swoole_async_read [swoole_async/read.phpt]
TEST 11/680 [PASS] swoole_async: swoole_async_read [swoole_async/readfile.phpt]
TEST 12/680 [PASS] swoole_async: recursive write file [swoole_async/recursive_write.phpt]
TEST 13/680 [SKIP] swoole_async: fd reuse [swoole_async/ref.phpt] reason: only in swoole debug version
TEST 14/680 [PASS] swoole_async: sequence copy 10m file [swoole_async/serial_read_copy_10m_file.phpt]
TEST 15/680 [PASS] swoole_async: swoole_async_set [swoole_async/set.phpt]
TEST 16/680 [PASS] swoole_async: swoole_async_write [swoole_async/write.phpt]
TEST 17/680 [PASS] swoole_async: swoole_async_read [swoole_async/writefile.phpt]
TEST 18/680 [PASS] swoole_atomic: add/sub/get/cmpset [swoole_atomic/atomic.phpt]
TEST 19/680 [PASS] swoole_atomic: multi wakeup [swoole_atomic/multi_wakeup.phpt]
TEST 20/680 [PASS] swoole_atomic: wakeup & wait [swoole_atomic/wait.phpt]
TEST 21/680 [PASS] swoole_atomic: wakeup & wait ex [swoole_atomic/wait_ex.phpt]
TEST 22/680 [PASS] swoole_buffer: read and write swoole_buffer [swoole_buffer/buffer_append.phpt]
TEST 23/680 [PASS] swoole_buffer: read and write swoole_buffer [swoole_buffer/buffer_clear.phpt]
TEST 24/680 [PASS] swoole_buffer: read and write swoole_buffer [swoole_buffer/buffer_expand.phpt]
TEST 25/680 [PASS] swoole_buffer: read and write swoole_buffer [swoole_buffer/buffer_read_write.phpt]
TEST 26/680 [PASS] swoole_buffer: read and write swoole_buffer [swoole_buffer/buffer_recycle.phpt]
TEST 27/680 [PASS] swoole_buffer: read and write swoole_buffer [swoole_buffer/buffer_substr.phpt]
TEST 28/680 [PASS] swoole_buffer: default contruct buffer [swoole_buffer/construct_buffer.phpt]
TEST 29/680 [PASS] swoole_channel: coro channel [swoole_channel/basic.phpt]
TEST 30/680 [PASS] swoole_channel: push & pop & stats [swoole_channel/pusu_pop_stats.phpt]
TEST 31/680 [PASS] swoole_client_async: big_package_memory_leak [swoole_client_async/big_package_memory_leak.phpt]
TEST 32/680 [PASS] swoole_client_async: onBufferFull & onBufferEmpty [swoole_client_async/buffer_full.phpt]
TEST 33/680 [PASS] swoole_client_async: connect & dns [swoole_client_async/connect_dns.phpt]
TEST 34/680 [PASS] swoole_client_async: connect refuse [swoole_client_async/connect_refuse.phpt]
TEST 35/680 [PASS] swoole_client_async: connect refuse with unix dgram
[swoole_client_async/connect_refuse_udg.phpt]
TEST 36/680 [PASS] swoole_client_async: connect refuse with unix stream
[swoole_client_async/connect_refuse_unix.phpt]
TEST 37/680 [PASS] swoole_client_async: connect_host_not_found [swoole_client_async/connect_timeout.phpt]
TEST 38/680 [PASS] swoole_client_async: connect twice [swoole_client_async/connect_twice.phpt]
TEST 39/680 [PASS] swoole_client_async: eof protocol [async] [swoole_client_async/eof.phpt]
TEST 40/680 [PASS] swoole_client_async: eof protocol [async] [close] [swoole_client_async/eof_close.phpt]
TEST 41/680 [PASS] swoole_client_async: getSocket debug [swoole_client_async/getSocket_bug.phpt]
```



□ RFC-1014 : TaskWorker 支持内置协程 已实现 接受 草案
#36 by matyhtf was closed 22 days ago

□ RFC-1012 Server->taskwait 支持协程调度 已实现 接受 草案
#28 by matyhtf was closed on 3 Aug 2018

□ RFC-1013 TaskWorker 支持异步操作和协程 已实现 接受 💬 1
#27 by huangzhhui was closed on 9 Oct 2018

□ RFC-1011 可设置是否在 Server 回调函数中自动创建协程 已实现 接受 草案 💬 1
#24 by matyhtf was closed on 29 Jun 2018

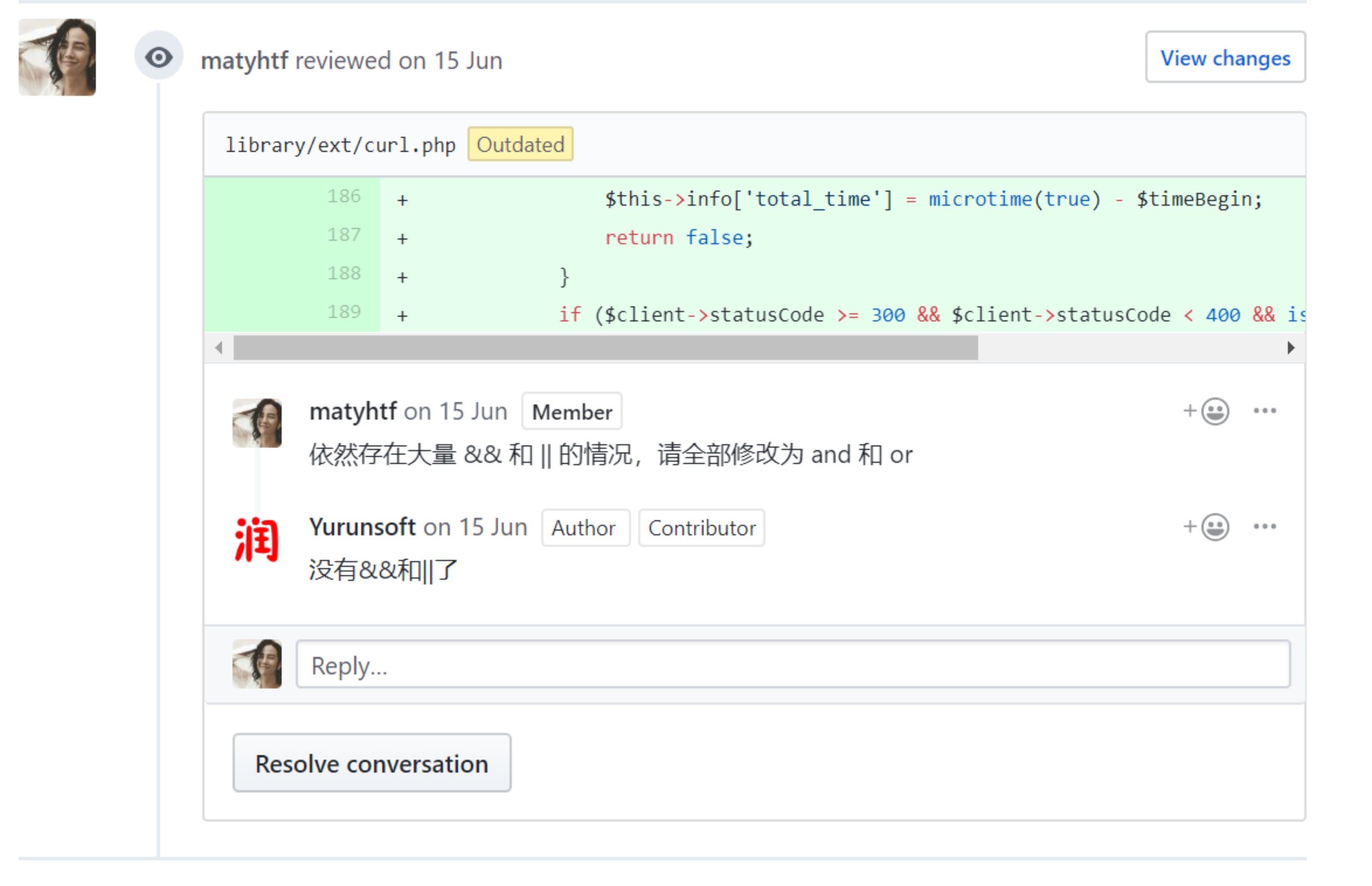
□ RFC-1010 server端的回调函数onReceive加一个参数标识请求到达server端的时间 已实现 接受 💬 2
#23 by lexin-arch was closed on 5 Nov 2018

□ RFC-1008 增加 Http\Resonse->redirect 方法 已实现 接受 💬 1
#17 by matyhtf was closed on 23 Apr 2018

□ RFC-1007 希望 Swoole\Http\Response 可以在别的进程中构造并使用 已实现 接受 💬 2
#15 by breath-co2 was closed on 23 Apr 2018

□ RFC-1006 Server/Client->send 支持协程化 已实现 草案
#13 by matyhtf was closed on 20 Apr 2018

□ RFC-1004 创建错误码对应的常量 已实现 接受
#10 by matyhtf was closed on 23 Mar 2018



matyhtf reviewed on 15 Jun

library/ext/curl.php

```
+                 $this->info['total_time'] = microtime(true) - $timeBegin;
+                 return false;
+             }
+             if ($client->statusCode >= 300 && $client->statusCode < 400 && is
```

matyhtf on 15 Jun

依然存在大量 && 和 || 的情况，请全部修改为 and 和 or

Yurunsoft on 15 Jun

没有&&和||了

Swoole 生态

Hyperf 框架

1. https://github.com/hyperf-cloud/hyperf

2. 在生产环境验证过的 PHP 协程框架

3. composer create-project hyperf/hyperf-skeleton

4. composer require hyperf/di

TarsPHP 微服务架构

1. https://tarsphp.gitbook.io/doc/tarsphp

2. 功能完善: 对标现有C++、JAVA、NodeJS体系功能

3. 灵活: 论灵活, 谁与PHP争锋?

4. 轻量: 用最轻量的设计, 点到即止, 即插即用

5. 高效: 插上SWOOLE协程的翅膀, 不得不飞

Swoole Tracker

1. https://www.swoole-cloud.com/index.html

2. 支持 PHP-FPM 和 Swoole 的服务器端 APM 完整解决方案

3. 包含：监控统计（stats）、链路追踪（trace）、性能分析（profile）、调试（debug）全套功能，提供 PHP & Swoole 的商业支持

4. 为使用 PHP 和 Swoole 的企业保驾护航

Swoole 专业开发者社区

1. 每周定期推送 专业 技术文章

2. PHP & Swoole 系列技术课程

3. PHP & Swoole 大牛技术分享

php-libreoffice 扩展

1. https://github.com/easysoft/phplibreoffice

2. 易软天创赞助开发的 PHP 高性能 office 操作扩展

3. 基于 libreoffice 库，性能极强，内存占用极少

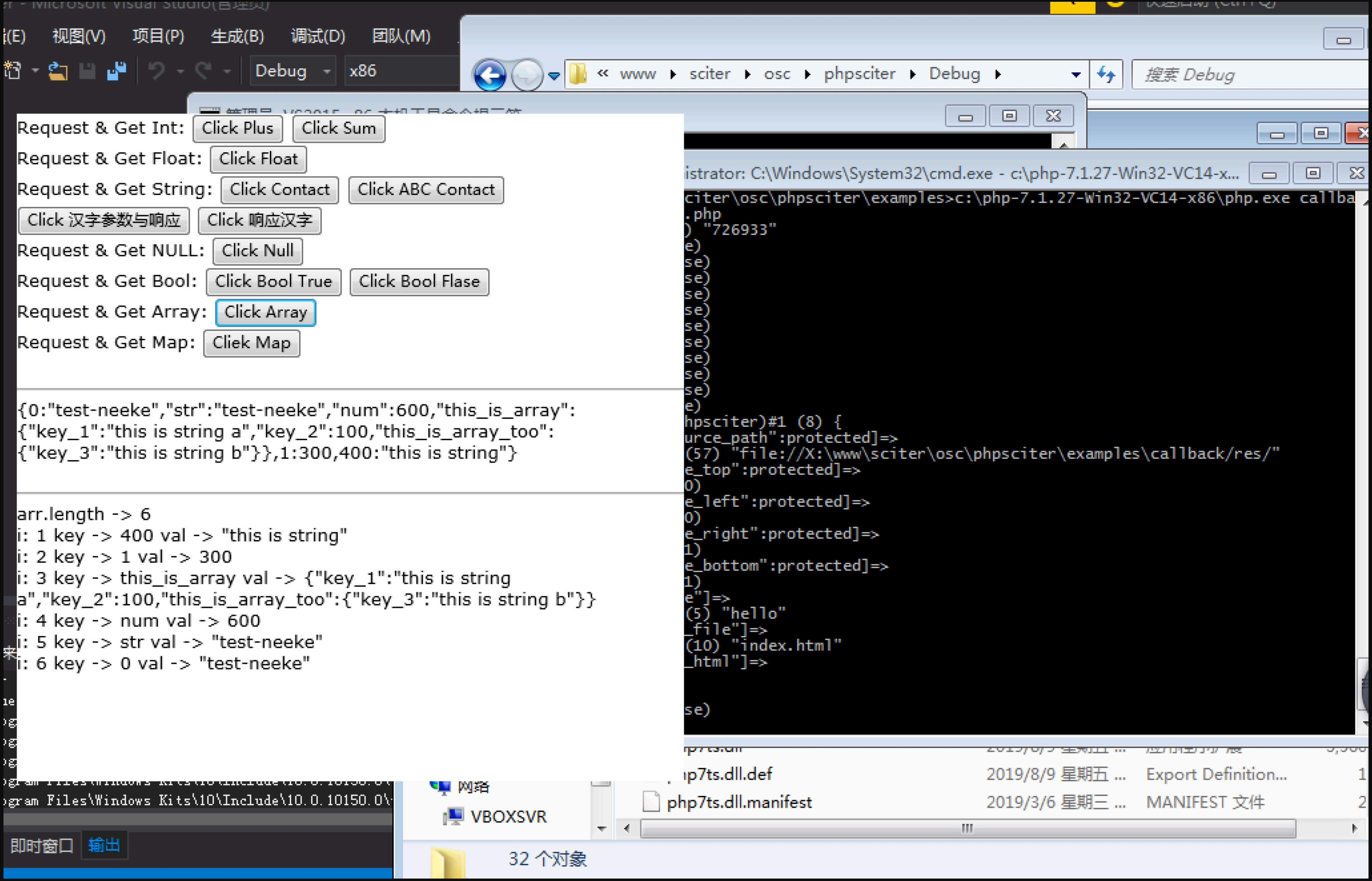
4. 高效实现 Word、PPT、Excel、PDF 文件操作

php-sciter 扩展

1. https://github.com/easysoft/phpsciter

2. 易软天创赞助开发的 PHP GUI 扩展

3. 基于 libsciter 库，使用 HTML5 + JS + PHP 编写桌面软件


(E) 视图(V) 项目(P) 生成(B) 调试(D) 团队(M)
Debug
x86
« www ▸ sciter ▸ osc ▸ phpsciter ▸ Debug ▸
搜索 Debug
Request & Get Int: Click Plus Click Sum
Request & Get Float: Click Float
Request & Get String: Click Contact Click ABC Contact
Click 汉字参数与响应 Click 响应汉字
Request & Get NULL: Click Null
Request & Get Bool: Click Bool True Click Bool Flase
Request & Get Array: Click Array
Request & Get Map: Cliek Map
istrator: C:\Windows\System32\cmd.exe - c:\php-7.1.27-Win32-VC14-x...
p7ts.dll.def 2019/8/9 星期五 ... Export Definition... 1
php7ts.dll.manifest 2019/3/6 星期三 ... MANIFEST 文件 2
网络
VBOXSVR
32 个对象
即时窗口 输出

```
{0:"test-neeke","str":"test-neeke","num":600,"this_is_array":
{"key_1":"this is string a","key_2":100,"this_is_array_too":
{"key_3":"this is string b"}},1:300,400:"this is string"}
```

```
arr.length -> 6
i: 1 key -> 400 val -> "this is string"
i: 2 key -> 1 val -> 300
i: 3 key -> this_is_array val -> {"key_1":"this is string
a","key_2":100,"this_is_array_too":{"key_3":"this is string b"}}
i: 4 key -> num val -> 600
i: 5 key -> str val -> "test-neeke"
i: 6 key -> 0 val -> "test-neeke"
```

```
citer\osc\phpsciter\examples>c:\php-7.1.27-Win32-VC14-x86\php.exe callba
.php
) "726933"
e)
se)
se)
se)
se)
se)
se)
se)
se)
se)
e)
hpsciter)#1 (8) {
urce_path":protected]=>
(57) "file://X:\www\sciter\osc\phpsciter\examples\callback/res/"
e_top":protected]=>
0)
e_left":protected]=>
0)
e_right":protected]=>
1)
e_bottom":protected]=>
1)
e"]=>
(5) "hello"
_file"]=>
(10) "index.html"
_html"]=>


se)
```

```
ogram Files\Windows Kits\10\Include\10.0.10150.0\
```

PHP 8 + Swoole 5

改进项	版本	意义
性能大幅提升	PHP 7.0	亿级 DAU 项目可用
Type Hint	PHP 7.4	强类型，更严谨，工程化
JIT	PHP 8.0	大幅提升密集计算性能

Swoole 4.4 LTS

1. 4.4 分支将作为长期维护版本（LTS），专注于 PHP 协程

2. 仅修复 BUG，不再引入新特性和重构

3. 尽可能地使用 PHP 实现 Library（内存安全、稳定、高效）

Swoole 5

- 成为工业级的 PHP 协程通信框架
- 健壮性是第一优先级
- 更专注，精益求精，严谨细致，打磨细节
- 十年磨一剑

- 支持 QUIC

- 支持 PSR-4、PSR-7 规范

- 支持分布式一致性协议

    A. zookeeper

    B. raft、paxos

招聘

- 连续 3 年 100% 以上增速，百亿级营收的项目

- 需要大量优秀的 PHP 工程师

- 参与构建学而思网校 PHP & Swoole 服务化架构

PHPCON 官方渠道：

官网：http://www.phpconchina.com

公众号：PHPCon

纪念品购买渠道：https://k.weidian.com/H3=4lVho

微信交流群：

添加个人微信号「PHPConChina」自动通过后，输入加群密码：11643 稍等自动拉群

谢谢