

Swoole 项目实践

@thinkpc

I 介绍主题

阿里云日志拉取

阿里云SLS日志拉取服务，多进程活跃监控

服务器管理代理

服务远程监控，控制及管理

实时监控

线上业务运行情况实时监控，实时分析

PHP标准服务容器

Websocket、HTTP、TCP统一服务容器

Websocket集群负载均衡

01

阿里云日志拉取

动态多进程管理，子进程心跳监控

阿里云日志抓取 – SDK不完善

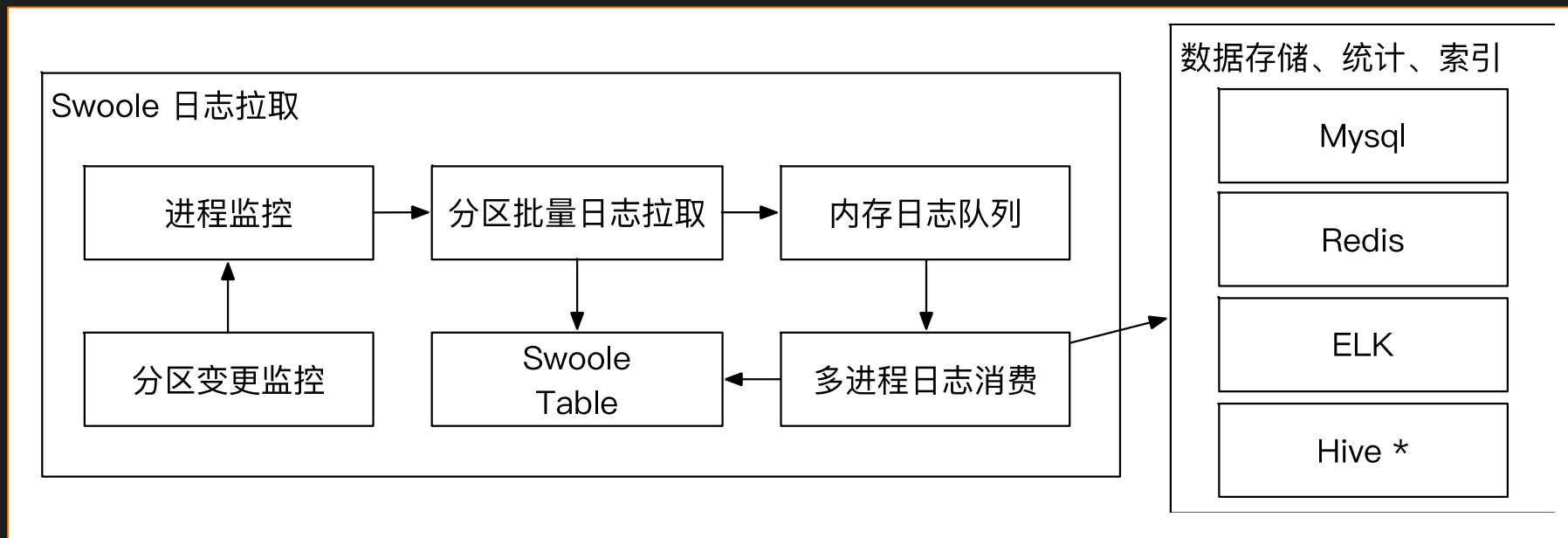
```
requestcore.class.php
View

0,0 @@ public function prep_request()
    curl_setopt($curl_handle, CURLOPT_MAXREDIRS, 5);
    curl_setopt($curl_handle, CURLOPT_HEADER, true);
    curl_setopt($curl_handle, CURLOPT_RETURNTRANSFER, true);
    curl_setopt($curl_handle, CURLOPT_TIMEOUT, 5104000);
    curl_setopt($curl_handle, CURLOPT_CONNECTTIMEOUT, 120);
    curl_setopt($curl_handle, CURLOPT_NOSIGNAL, true);
    curl_setopt($curl_handle, CURLOPT_REFERER, $this->request_url);
    curl_setopt($curl_handle, CURLOPT_USERAGENT, $this->useragent);

610      curl_setopt($curl_handle, CURLOPT_MAXREDIRS, 5);
611      curl_setopt($curl_handle, CURLOPT_HEADER, true);
612      curl_setopt($curl_handle, CURLOPT_RETURNTRANSFER, true);
613      + curl_setopt($curl_handle, CURLOPT_TIMEOUT, 20);
614      + curl_setopt($curl_handle, CURLOPT_CONNECTTIMEOUT, 5);
615      curl_setopt($curl_handle, CURLOPT_NOSIGNAL, true);
616      curl_setopt($curl_handle, CURLOPT_REFERER, $this->request_url);
617      curl_setopt($curl_handle, CURLOPT_USERAGENT, $this->useragent);
```

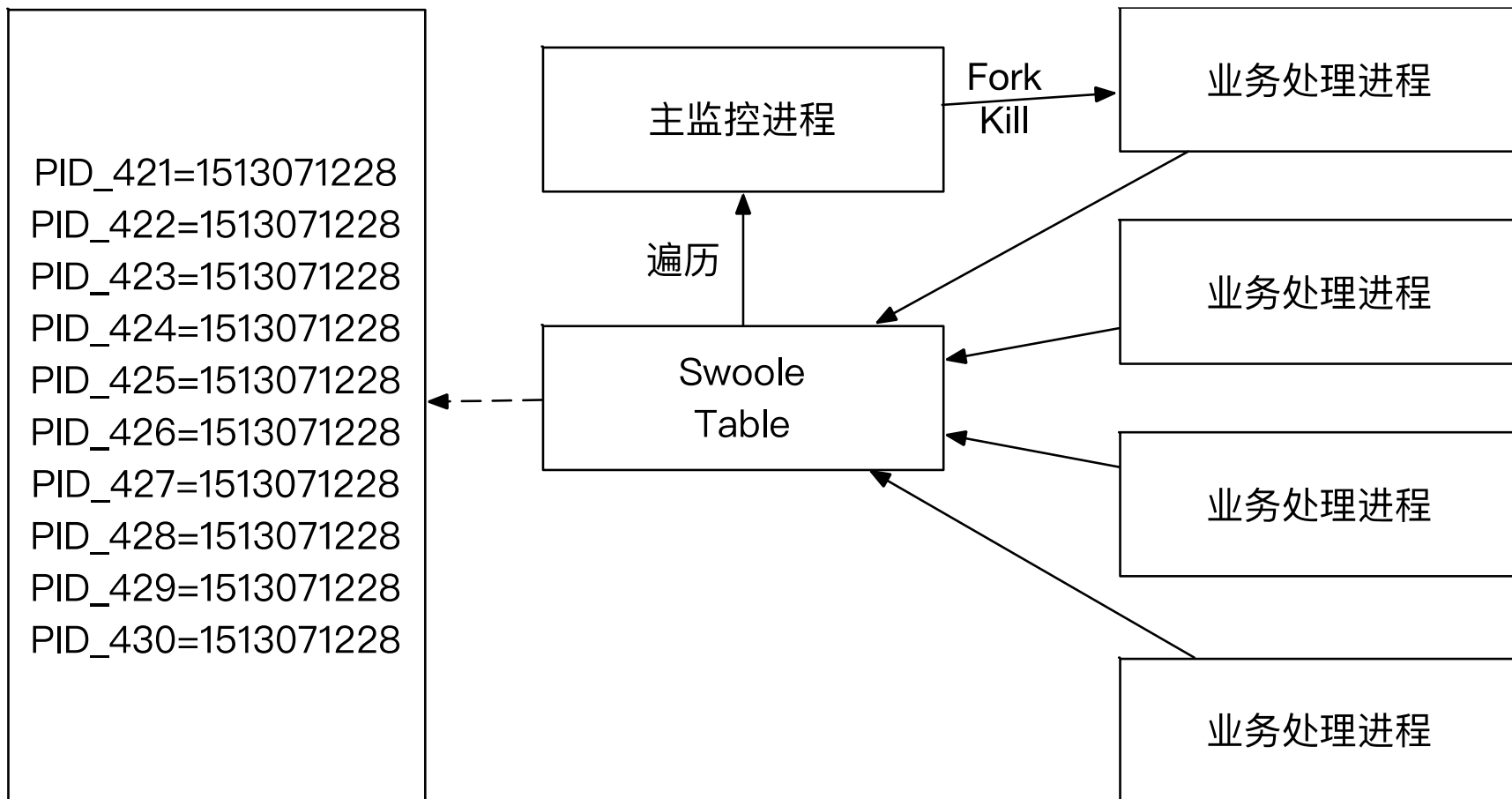
日志库是多Shards 但是官方只有单进程通讯SDK
SDK底层使用Keep Alive节省资源
触发Curl隐含bug导致进程卡死

阿里云日志抓取 – 整体结构



多进程并行批量拉取Shard日志
每天500w条日志

阿里云日志抓取 - 子进程健康检查



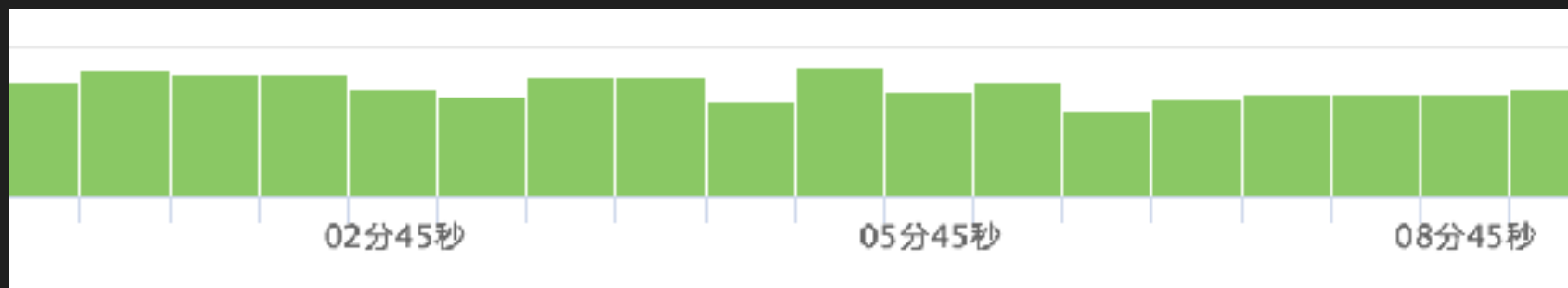
`timestamp <= time() + 10 second` 检测卡死

02

实时监控

线上业务运行情况实时监控，实时分析

实时监控 – 统计粒度



- 大型系统QPS计算粒度不可能是按秒的
- 需要确定最小粒度
- 常见为30秒

实时监控-时间块划分算法

统计窗口
3 分钟一个块

$$\text{Index} = (\text{time}() - \text{strtotime}(\text{"Y-m-d"})) / 180 \text{ Second}$$

Time
Block

Time
Block

Time
Block

Time
Block

Time
Block

03

服务器节点管理

多服务器情况自动上报，服务监控，远程命令管理

服务节点管理

服务分组1~n

NodeAgent 及 服务器 1

文件传输

服务发现及配置落地

进程管理

脚本执行

其他自定义功能

NodeAgent 及 服务器 2~n

文件传输

服务发现及配置落地

进程管理

脚本执行

其他自定义功能

exec的梗

```
$proc = new \swoole_process(function (\swoole_process $proc) use ($cmd, $param) {  
    $proc->exec($cmd, $param);  
    echo PHP_EOL . "-----end-----" . PHP_EOL;  
}, true);  
  
//if the new version of process  
if (method_exists($proc, method_name: "setTimeout")) {  
    //read timeout is 5  
    $proc->setTimeout(5.0);  
}  
$pid = $proc->start();  
$output = $proc->read();
```

Process->exec read timeout
tail -f 命令

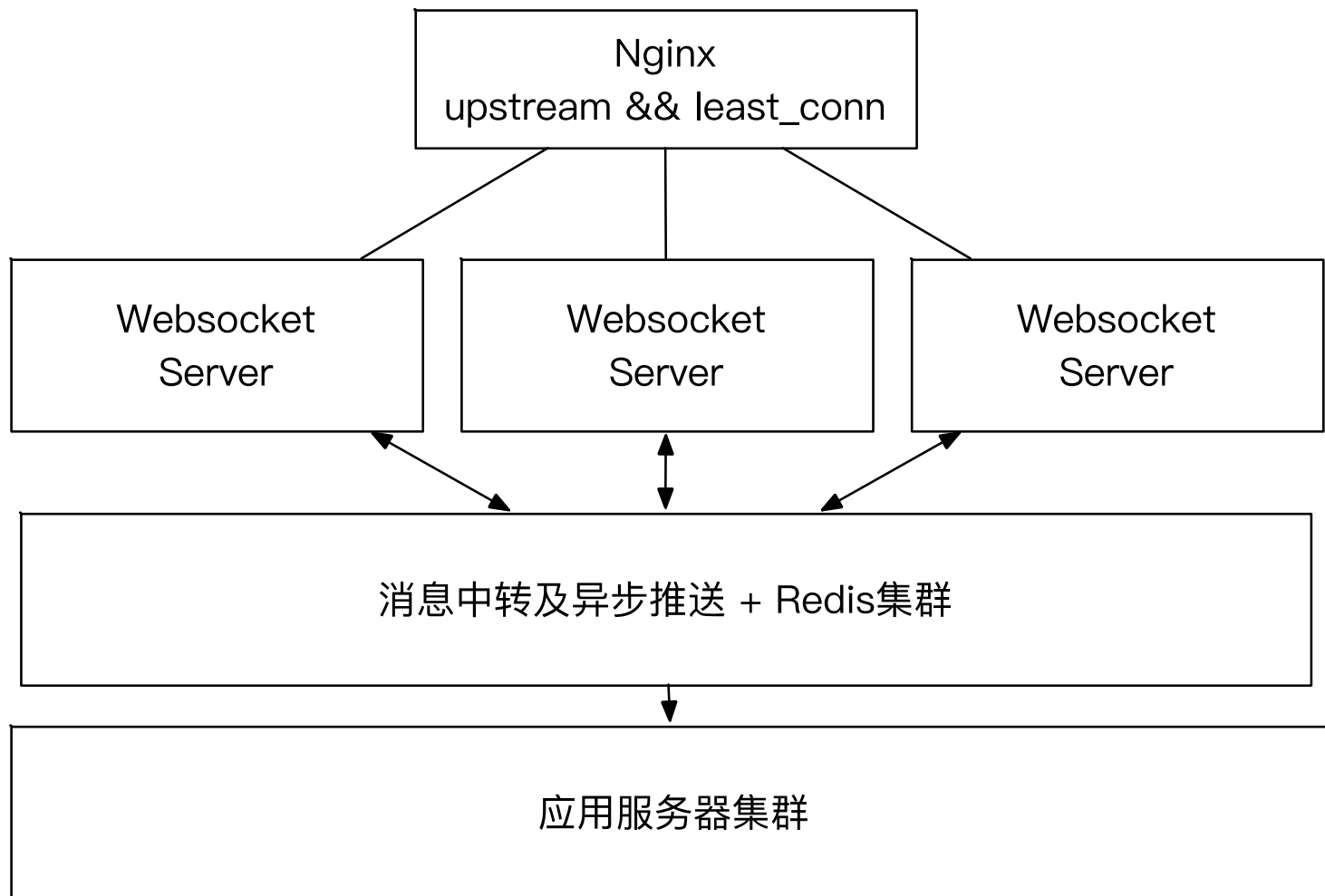
04

服务容器

多协议支持，统一路由，长链接层消息流转

Websocket服务

Websocket长链接服务



基于连接数 负载均衡

```
upstream stream_backend {  
    least_conn;  
  
    server backend1.example.com:12345 weight=5;  
    server backend2.example.com:12345;  
    server backend3.example.com:12346;  
}
```

Websocket路由

ws://websoket.xxx.com/chatroom/live

- onOpen
- onMessage
- onRequest
- onClose 注：需要区分http、Websocket

消息推送

ws://websocket.xxx.com/chatroom/push

- WebSocket支持http请求
- FD是共享的
- 接收到消息验证来源后，直接推送

Q & A