

Node性能跟踪 与稳定性优化

张轩丞（朋春）

pengchun@alibaba-inc.com

数据中间层

- http协议，SQL接口
- 后端接mysql、hbase、webservice...
- 接近1亿请求/天，平均20ms响应
- 4台机器，cpu idle > 90%

选择时间: 昨天 最近7天

全站搜索关键词排行 类目分布与趋势 关联热词 关联宝贝

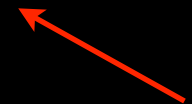
每页显示 50 条 搜索:

关键词	搜索次数	占比	点击次数	商城点击占比	搜索人数	点击率	当前宝贝数	转化率	直通车
1 牛仔裤	3,862,372	40.29%	1,261,258	21.29%	1,141,179	32.66%	5,568,376	0.40%	1.52
2 牛仔裤 女	687,384	7.17%	318,407	25.17%	171,797	46.32%	1,724,751	0.57%	1.16
3 牛仔裤 男	401,775	4.19%	209,912	31.87%	104,241	52.25%	2,105,850	1.25%	2.67
4 牛仔裤 女 韩版 潮	214,300	2.24%	96,271	18.18%	56,447	44.92%	104,462	0.49%	0.84

```

SELECT r.query, search_num, ..., s.auction_num
FROM mysql.rpt_query_effect_d r
INNER JOIN taobao.search s
ON r.query = s.query WHERE ...
ORDER BY search_num DESC LIMIT 5

```



方法与武器

- console.time、console.timeEnd
- benchmark.js
- v8-profile
- top、iotop、strace、lsof、perf...
- 监控工具

性能分析与调优



A screenshot of a Weibo profile for user 'flyinweb (占超群)'. The profile features a cartoon avatar of a person with glasses and a beard. The background image shows a desert landscape with red rock formations. The profile information includes the user's name, a level indicator (LV 3), and a URL. The bio mentions being a male, heterosexual, from Beijing's Chaoyang District, and working at Alibaba. The tags list various technical and personal interests. The statistics show 372 followers, 1186 fans, and 1213 tweets. The interface includes buttons for mutual follow and chat.

flyinweb (占超群) LV 3 <http://weibo.com/317347888>

男 · 异性恋 · 北京 朝阳区 · 公司 淘宝网

标签: 数据库 · 实时计算 · 电影控 · OSGI · 架构设计 · nodejs · 数据挖掘

简介: 技术爱好者, 略懂数据, 资深影迷, 电子商务新兵。 [更多资料»](#)

372 关注 | 1186 粉丝 | 1213 微博

[互相关注 | 取消](#) [聊天](#)

<http://www.slideshare.net/flyinweb/nodejs2011121820>

内核参数

```
[pengchun]$ cat /etc/sysctl.conf
```

```
...
```

```
net.ipv4.tcp_max_syn_backlog = 65536
net.ipv4.tcp_timestamps = 0
net.ipv4.tcp_tw_recycle = 1
net.ipv4.tcp_tw_reuse = 1
net.ipv4.ip_local_port_range = 1024 65535
net.ipv4.tcp_fin_timeout = 30
net.ipv4.tcp_keepalive_time = 180
net.ipv4.tcp_max_tw_buckets = 5000
```

内核参数

- `ulimit -n 65535`
- `ulimit -c unlimited`

响应时间

- 统计周期内均值
- 响应时间分布

关注gc

```
$ node --trace_gc --trace_gc_nvp \
  app.js &> gc.log &
$ tail gc.log
```

```
5270 ms: pause=1 mutator=38 gc=ms external=0 mark=0 sweep=1 sweepns=0 evacuate=0
new_new=0 root_new=0 old_new=0 compaction_ptrs=0 intrac
ompaction_ptrs=0 misc_compaction=0 total_size_before=4213000 total_size_after=3181224
holes_size_before=79136 holes_size_after=1773112 alloc
ated=1096896 promoted=90968 stepscount=1 stepstook=5
```

```
5297 ms: pause=1 mutator=25 gc=ms external=0 mark=0 sweep=1 sweepns=0 evacuate=0
new_new=0 root_new=0 old_new=0 compaction_ptrs=0 intrac
ompaction_ptrs=0 misc_compaction=0 total_size_before=3956552 total_size_after=3292576
holes_size_before=152072 holes_size_after=1659160 allo
cated=775328 promoted=144040 stepscount=1 stepstook=5
```

```
$ ./deps/v8/tools/gc-nvp-trace-
processor.py ./gc.log
```

进程管理

- aleafs' pm

- 请求分发，利用多CPU处理 (cluster)
- 子进程容灾，自动重启 (forever)
- 系统信号的处理
- 轻量，接口简单

<https://github.com/aleafs/pm>

异常捕获

```
var logException = function (e) {  
    // write stack to error log  
};
```

软件异常

异常名称	异常次数	异常详情	异常时间
OTSSStorageServerBusyErrorException	1	详细	2012-09-13 16:00
OTSSStorageTxnLockKeyFailErrorEx			
HbaseErrorException			
RequestTimeoutErrorException			

异常详细内容

2012-09-13 09:52:09.000 nodejs.RequestTimeoutErrorException: RequestTimeoutError: Request Timeout after 10010m
at createHangUpError (http.js:1253:15)
at Socket.socketCloseListener (http.js:1304:23)
at Socket.EventEmitter.emit (events.js:115:20)
at Socket._destroy.destroyed (net.js:358:10)
at process.startup.processNextTick.process._tickCallback (node.js:244:9)
URL: undefined
Data: {"query":"{\\"sql\\":\\"UPDATE hbase_tcif.tcif SET cf1:member_car_info=:member_car_info WHERE row = :row\\",\\"}

异常捕获

```
iocall(..., function (err, res) {  
  err && logException(err);  
});
```

异常捕获

```
ioCall(..., function (err, res) {  
  err && logException(err);  
});
```

```
try{...} catch (e) {  
  logException(e);  
}
```

异常捕获

```
iocall(..., function (err, res) {  
    err && logException(err);  
});
```

```
try{...} catch (e) {  
    logException(e);  
}
```

```
process.on('uncaughtException',  
    function (e) {  
        logException(e);  
        process.exit(1);  
    });
```

容灾测试

- netblackhole
 - 超时控制是否合理
 - 强弱依赖是否恰当

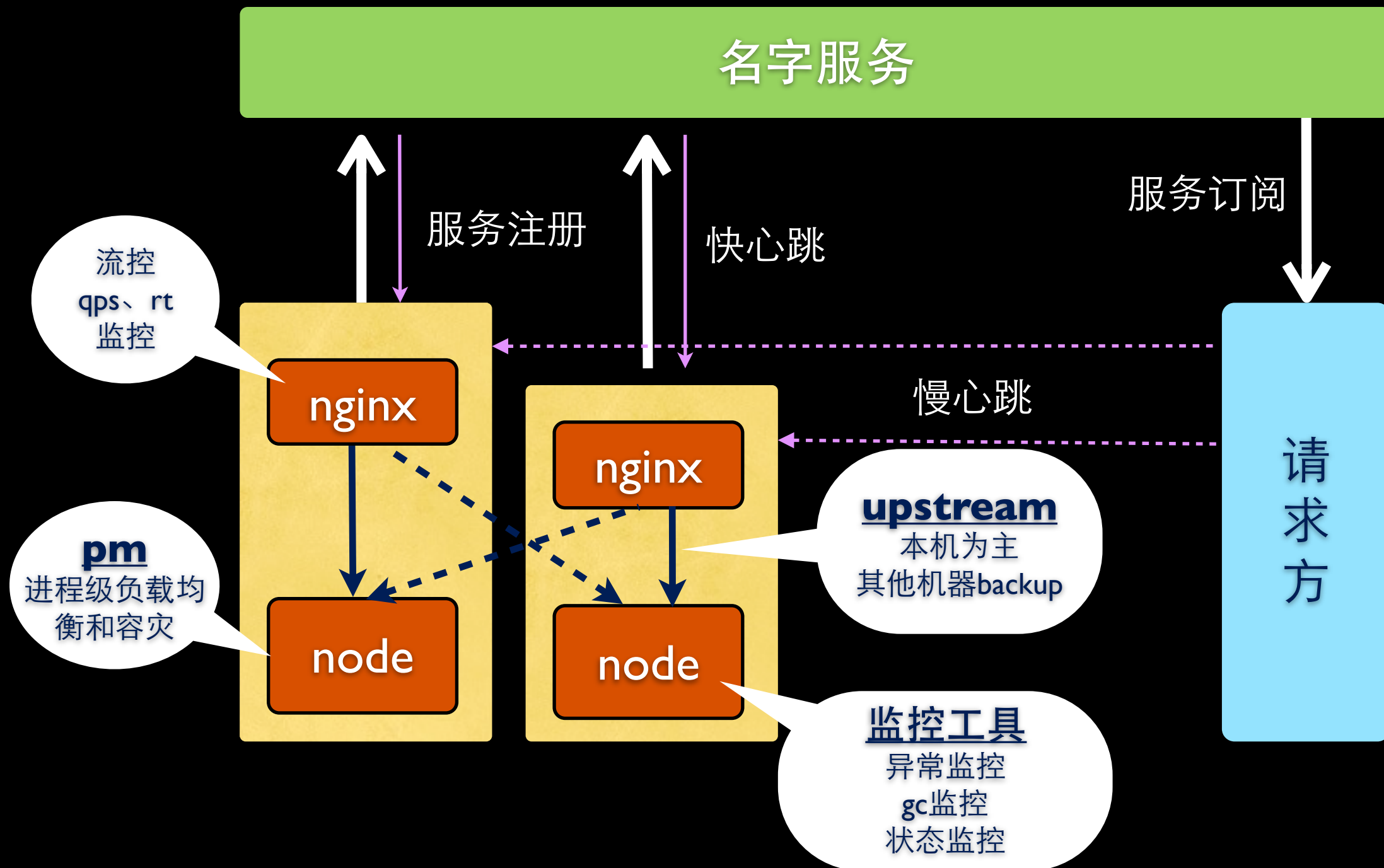
```
/**
 * @ 模拟一个异常的服务
 * @ 测试Client对异常服务的容灾
 */
var s =
require('netblackhole').create(1234);

s.never_response();

Client.init(['localhost:6379',
'localhost:1234']);
```

<https://github.com/aleafs/netblackhole>

高可用架构



一家之言

- 节省内存，规避gc
- “满载”是指跑满CPU
- 所有的“错误”都应该被关注
- 稳定出自监控



关于作者

- work at 阿里巴巴
- 新浪微博: @我是aleafs
- Github: <https://github.com/aleafs>
- zhangxc83@gmail.com