支撑百亿级流量PHP引擎HHVM在百度的实战

huzhiguang@baidu.com
weibo:huzhiguang88
blog: saiyaren.iteye.com

自我介绍

- 胡志广（百度-基础架构部）
- 2013 年加入百度 （之前就职于2家创业公司和京东）
- 负责方向
  - HHVM
  - 百度私有云机器管理
- Facebook 感谢

assist with pull requests. Particular thanks to **Daniel Sloof**, **Markus Staab**, **Vadim Borodavko**, **Kristaps Kaupe**, **huzhiguang**, and the many others in the community currently pushing commits to our source base.

http://hhvm.com/blog/875/wow-hhvm-is-fast-too-bad-it-doesnt-run-my-code

目录

- 为什么使用HHVM
- HHVM in baidu
- HHVM VS PHP7

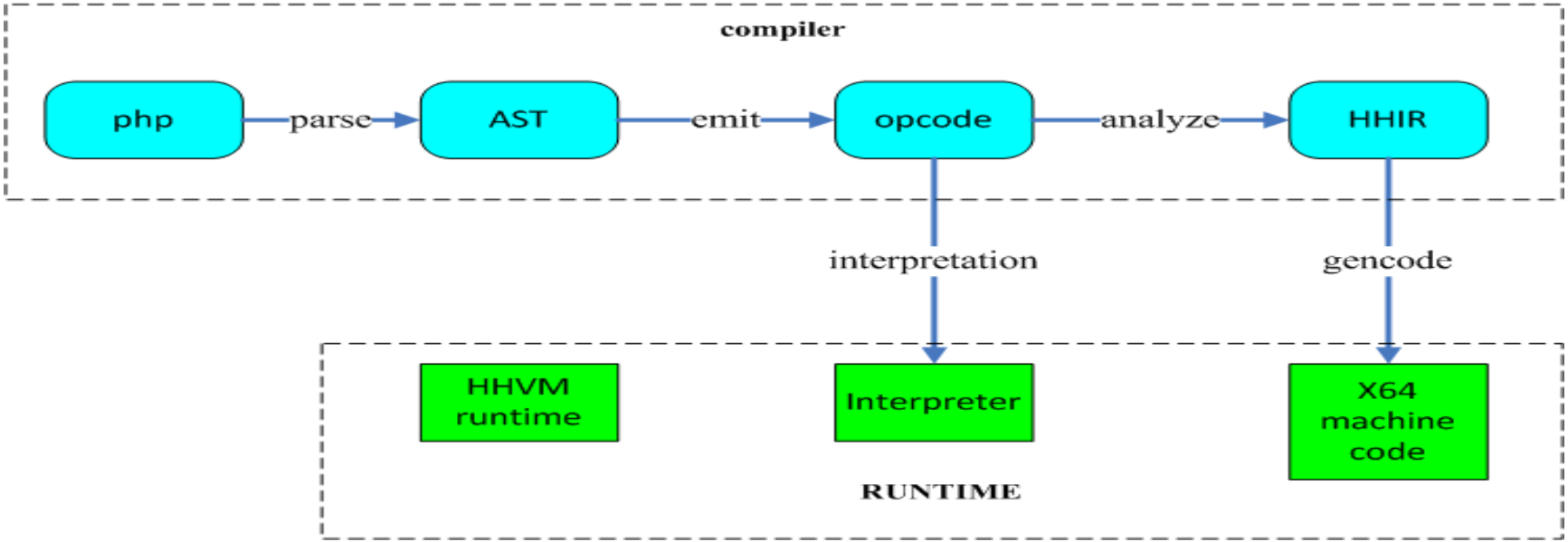
为什么使用HHVM

- HHVM 是什么？
- HHVM 百度调研

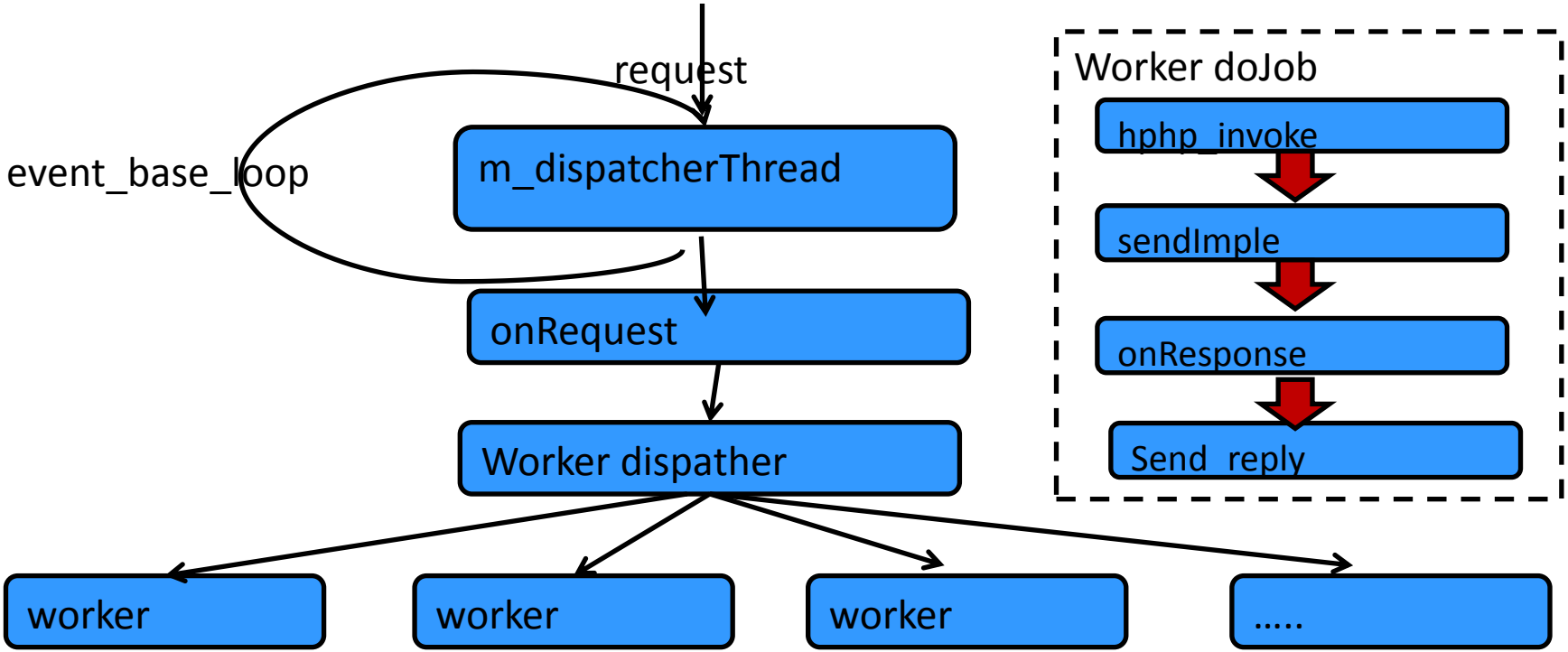
HHVM 是什么？

- PHP
- Hiphop
- HHVM

HHVM流程


compiler
php
parse
AST
emit
opcode
analyze
HHIR
interpretation
gencode
HHVM runtime
Interpreter
X64 machine code
RUNTIME

HHVM Server 模型



HHVM admin server & trace

Admin server

```
/stop:                stop the web server
    instance-id       optional, if specified, instance ID has to match
/translate:           translate hex encoded stacktrace in 'stack' param
    stack             required, stack trace to translate
    build-id          optional, if specified, build ID has to match
    bare              optional, whether to display frame ordinates
/build-id:            returns build id that's passed in from command line
/instance-id:         instance id that's passed in from command line
/compiler-id:         returns the compiler id that built this app
/repo-schema:         return the repo schema id used by this app
/check-repo:          check tables of hhbc to judge whether error or not
/check-load:          how many threads are actively handling requests
/check-queued:        how many http requests are queued waiting to be
                      handled
/check-health:        return json containing basic load/usage stats
/check-ev:            how many http requests are active by libevent
/check-backlog:       how many http requests in listen queue
/check-pl-load:       how many pagelet threads are actively handling
                      requests
/check-pl-queued:     how many pagelet requests are queued waiting to
                      be handled
/check-mem:           report memory quick statistics in log file
/check-sql:           report SQL table statistics
/check-sat            how many satellite threads are actively handling
                      requests and queued waiting to be handled
/status.xml:          show server status in XML
/status.json:         show server status in JSON
/status.html:         show server status in HTML
/stats-on:            main switch: enable server stats
/stats-off:           main switch: disable server stats
/stats-clear:         clear all server stats
/stats-web:           turn on/off server page stats (CPU and gen time)
/stats-mem:           turn on/off memory statistics
/stats-apc:           turn on/off APC statistics
/stats-apc-key:       turn on/off APC key statistics
/stats-mcc:           turn on/off memcache statistics
/stats-sql:           turn on/off SQL statistics
/stats-mutex:         turn on/off mutex statistics
    sampling          optional, default 1000
/stat.keys:           list all available keys
    from              optional, <timestamp>, or <-n> second ago
    to                optional, <timestamp>, or <-n> second ago
/stats.xml:           show server stats in XML
    from              optional, <timestamp>, or <-n> second ago
    to                optional, <timestamp>, or <-n> second ago
    agg               optional, aggragation: *, url, code
    keys              optional, <key>,<key/hit>,<key/sec>,<:regex:>
    url               optional, only stats of this page or URL
```


```
{
   "load":0
,  "queued":0
,  "hhbc-roarena-capac":0
,  "tc-size":0
,  "tc-stubsize":668
,  "targetcache":5224
,  "rds":5224
,  "units":0
,  "Funcs":4272
,  "EvaledUnits":0
,  "CreateFuncs":0
}
```


Trace

```
Reentry: enter (0x7fdd72c7ffc0) from top-level
dispatch: Enter ExecutionContext::dispatch(0x7fdd72c7ffc0)
dispatch: 0: FPushFuncD
dispatch: 6: FCall
dispatch: Halt ExecutionContext::dispatch(0x7fdd72c7ff90)
------------------------- after initial translation -------------------------
digraph G {
B0 -> B2; B0 -> B1
}

 B0:
Function main at 39 (ID 2)
    --- bc 39, spoff 0 (main)
  39: FPushFuncD 2 "max2"
    (00) t0:FramePtr = DefFP
    (01) t1:StkPtr = DefSP<0> t0:FramePtr
    (04) CheckCold<0> -> B1<Unlikely>
  -> B2

 B2: (preds B0)
    (08) t5:StkPtr = SpillFrame<2> t1:StkPtr, t0:FramePtr, Func(max2), InitNull
    --- bc 59, spoff 5 (main)
  59: FCall 2
    (12) t9:StkPtr = Call<> t5:StkPtr, 22, Func(max2), "wxy", "abc"

    --- bc 61, spoff 1 (main)
  61: UnboxR
    (13) SyncABIRegs t0:FramePtr, t9:StkPtr
    (14) ReqBindJmp<61>
  no fallthrough

----------------------unlikely blocks----------------------
 B1<Unlikely>: (preds B0)
    --- bc 39, spoff 0 (main)
  39: FPushFuncD 2 "max2"
    (02) SyncABIRegs t0:FramePtr, t1:StkPtr
    (03) ReqRetranslateOpt<0, 39>
  no fallthrough
-------------------------------------------------------------------------------
```

HHVM 百度调研（2013）

- HHVM in JD
  - Cpu 节约**57%**
  - 响应时间**28.5ms ->15ms**
- BAIDU 某业务线调研
  - Beachmark

引擎	bench.php 耗时	micro_bench.php 耗时	bench_third.php 耗时
php5.2	6.692s	41.890s	9.226s
php5.5	3.609s	14.972s	5.893s
hhvm	0.579s	5.832s	2.869s

实际业务

引擎	优化前	业务优化后
php5.2	127ms	103ms
php5.5	107ms	87ms
hhvm	72ms	26ms

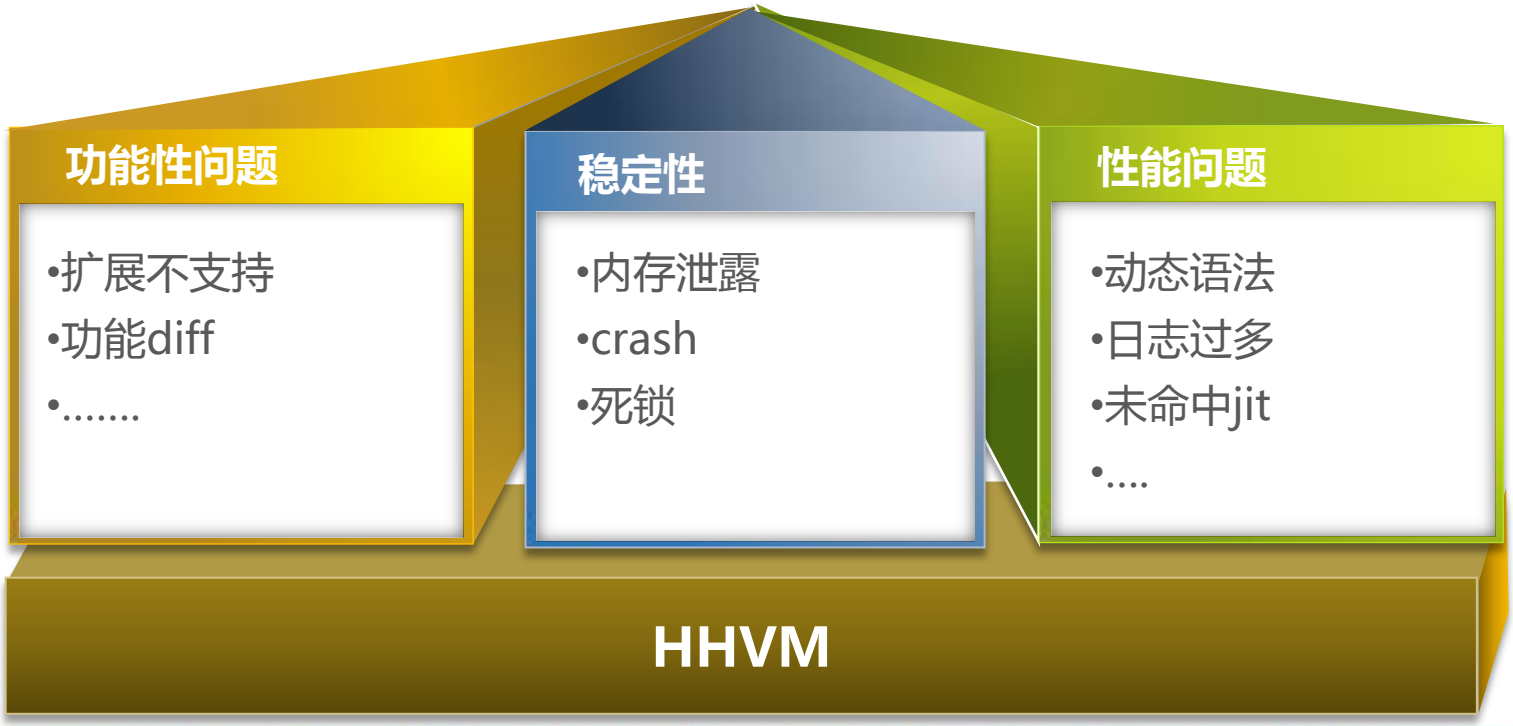
HHVM in Baidu

- HHVM 上线效果
- HHVM 使用问题
- HHVM 生态
- HHVM 迁移方案
- HHVM baidu 优化
- 线上问题分析
- Baidu 对hhvm的贡献

HHVM 线上效果

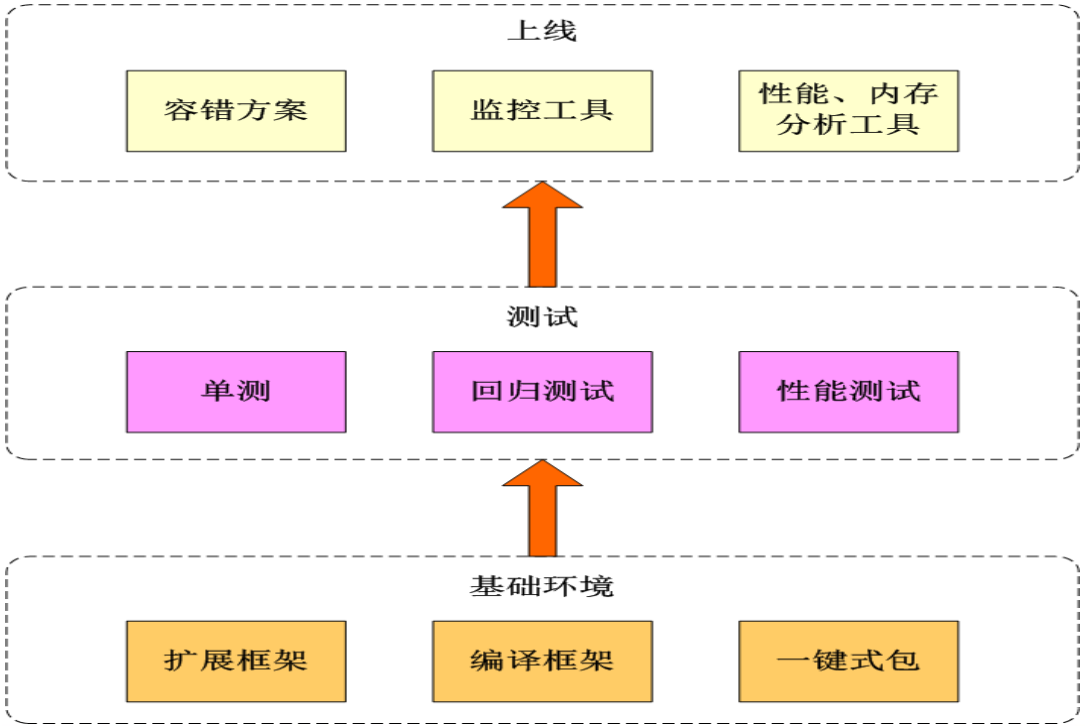
- 部署机器规模超过**6000**台+
- 日均访问HHVM的PV近**千亿**
- CPU使用率节约**40%~60%**
- 响应时间减少**50%~80%**

HHVM 使用问题


功能性问题
•扩展不支持
•功能diff
•.....
稳定性
•内存泄露
•crash
•死锁
性能问题
•动态语法
•日志过多
•未命中jit
•....
HHVM

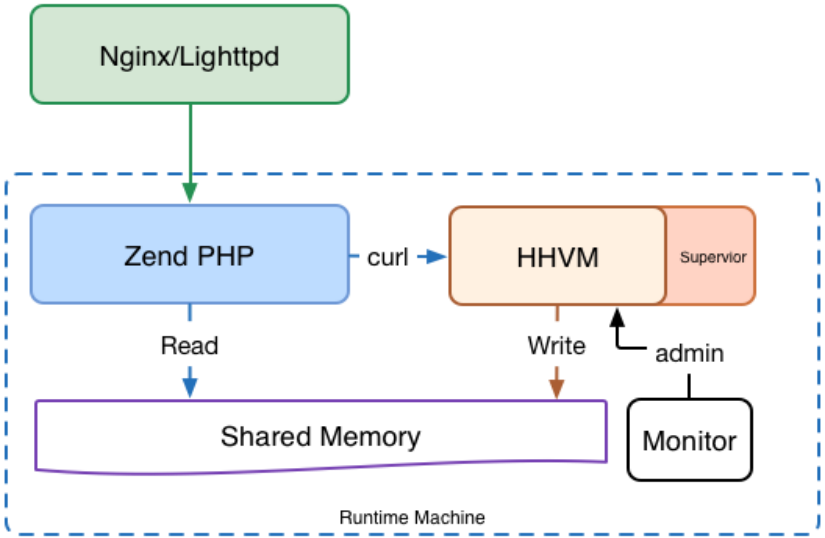
HHVM 生态

- 扩展支持**30+**
- HHVM 团队
- 内部开源

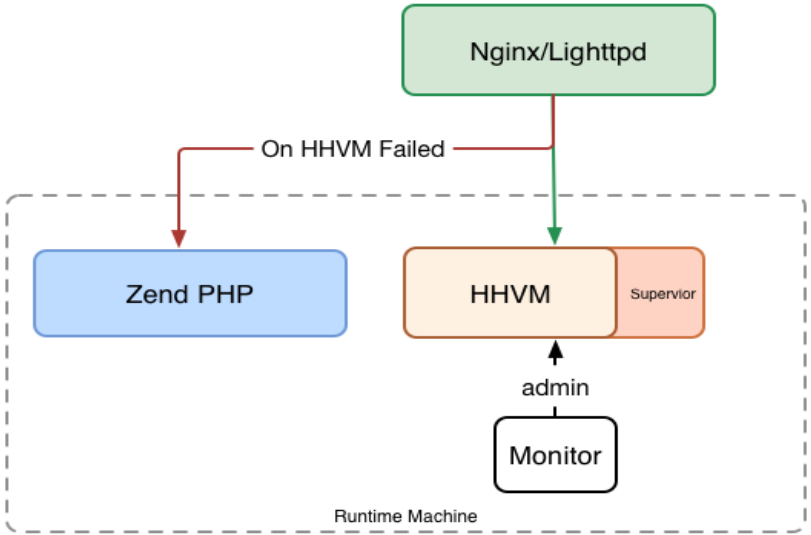


HHVM 迁移方案

旁路



全量



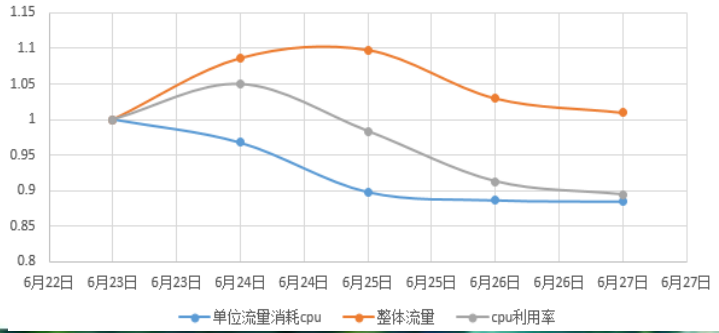
HHVM baidu 优化

- 已优化：
  - File Io
  - 更新文件锁冲突优化
- 待优化：
  - Fastcgi 长连接
  - jit gc
  - 引擎级别协程
  - 更多IO、性能优化

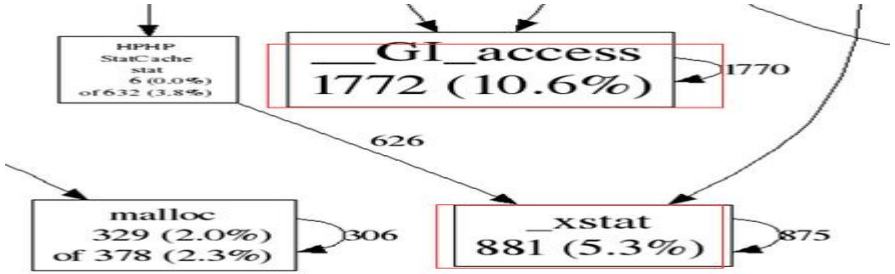
HHVM Baidu 优化-File io

线上某业务效果

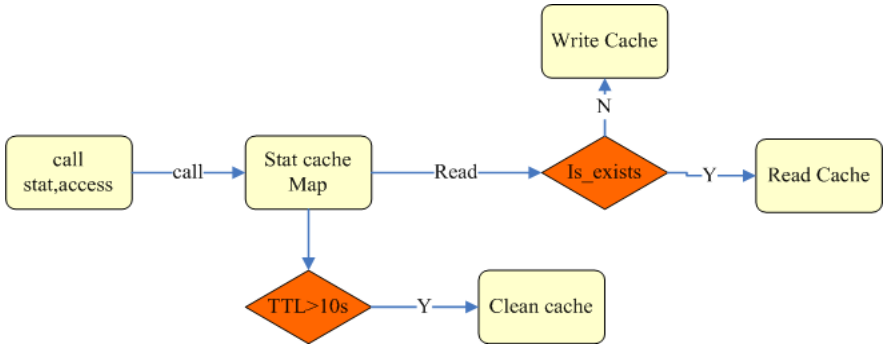
- 节约了cpu减少**10.6%**
- 响应时间减少减少**11.5%**
- 折合机器节约**329**台



分析某业务线HHVM 调用

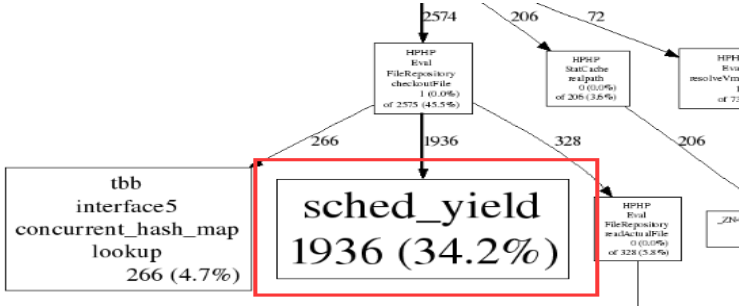


优化流程

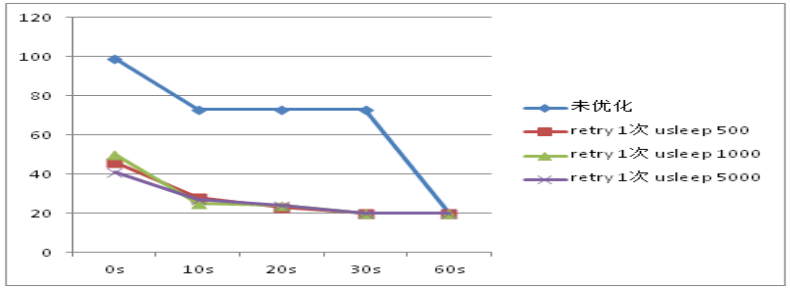


HHVM baidu 优化-上线CPU优化

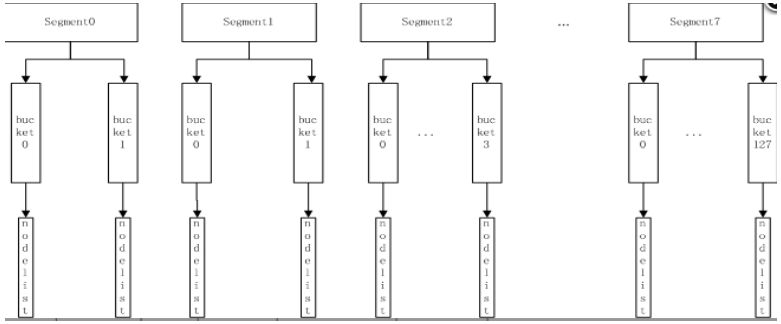
问题定位



优化效果



TBB Map 分析




```
if( !result->try_acquire( n->mutex, write ) ) {
    for( tbb::internal::atomic_backoff backoff(true);; ) {
        if( result->try_acquire( n->mutex, write ) ) break;
        if( !backoff.bounded_pause() ) {
            // the wait takes really long, restart the operation
            b.release();
            __TBB_ASSERT( !op_insert || !return_value, "Can't acquire new item in locked bucket?" );
            __TBB_Yield();
            m = (hashcode_t) itt_load_word_with_acquire( my_mask );
            goto restart;
        }
    }
}
```

HHVM 线上问题分析

- CPU 异常
- 内存异常
- 耗时异常

HHVM 线上问题分析

- 可复现
  - cpu :gperftool
  - mem:jemalloc
  - 耗时：strace、xhprof
- 不可复现
  - 监控分析
  - Tcp copy线下分析
  - 线上profiling（风险大）

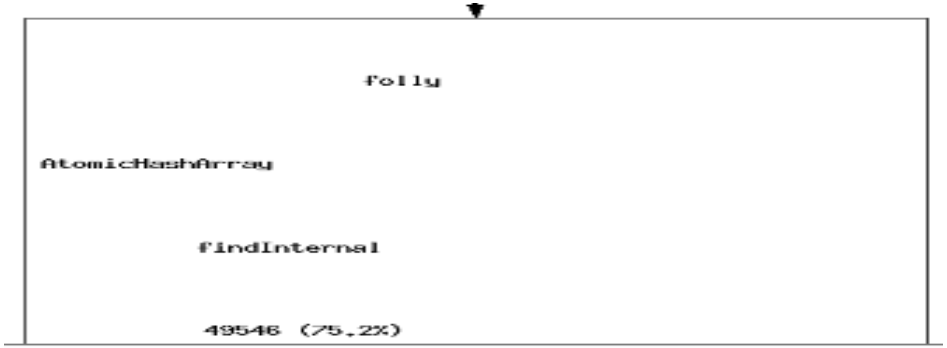
HHVM 线上分析案例-1

1. 问题：CPU高、响应速度慢
2. 分析1：hhvm jit full core stack
3. 分析2：监控unit+func 线程增长
4. 定位：eval动态语法问题
5. 优化：600ms->0.8ms cpu 恢复正常

具体分析见：http://lamp.baidu.com/2014/10/17/dong-tai-yu-fa-de-xing-neng-wen-ti-fen-xi/

HHVM 线上分析案例2

- 问题：Cpu 打满
- 定位：folly Map 冲突位占满
- 解决：调整folly Map 初始化大小




```
insertInternal(KeyT key_in, T&& value) {
  ......
  if (isFull_.load(std::memory_order_acquire))
    return false; //满了，不让再插入这个map了

  ++numEntries_; //已插入的数量
  if (numEntries_.readFast() >= maxEntries_) {
    isFull_.store(true, std::memory_order_relaxed); //isfull设置为true
  ......
}
```

Baidu 对hhvm的贡献

- 贡献社区：解决问题、回馈源码
- 与facebook hhvm owner多次邮件交流
- 帮助国内许多公司解决使用HHVM问题
- 建设lamp.baidu.com站交流

HHVM VS PHP7-PHP7 优化

- zval 优化(sizeof 24->16 bytes)
- Hash table 优化 (链式->开放式）
  - HashTable 从72 减少到 56 bytes
  - Bucket size 从72 减少到32 bytes
  - 减少cpu miss
- 内存管理模式优化
  - 改了了类似jemalloc 动态管理模式，提升cpu cache
- 强类型支持
- 编译加入AST

- **-注:上面优化均针对php5**

HHVM VS PHP7

HHVM	PHP7
typeValue(16)	Zval(16)
Hashtable(开放式）	Hashtable（开放式）
内存管理jemalloc	类似jemalloc的内存模型
强类型(hack)	强类型
AST (hiphop 时就支持）	AST
Jit	主线版无
Async 语法	无（可用swoole寻找代替方案）
Opcode持久化（repo）	无
多线程、支持http server、admin server	多进程

HHVM VS PHP7 ext

```
<?php
    echo hello("hello");
```

HHVM

```
<<__Native>>
function hello(string $str):string;
```

```
        virtual void moduleInit() {
                HHVM_FE(hello);

                loadSystemlib();
        }
```

```
String HHVM_FUNCTION(hello,const String & str) {
    return str;
}
```

PHP7

```
*/
const zend_function_entry hello_functions[] = {
    PHP_FE(hello,    NULL)        /* For testing, remove later. */
    PHP_FE_END  /* Must be the last line in hello_functions[] */
};
```

```
PHP_FUNCTION(hello)
{
        char *str = NULL;
        size_t str_len, len;

        if (zend_parse_parameters(ZEND_NUM_ARGS(), "s", &str, &str_len) == FAILURE) {
            return;
        }

        zend_string *strg;
        strg = strpprintf(0, str);
        RETURN_STR(strg);
}
```

参考资料

- PHPNG a New Core for PHP7 @Dmitry Stogov
- The Secret of PHP7’s Performance @Laruence
- The HipHop Virtual Machine @Facebook
- How Facebook’s HHVM Uses C++ for Fun and Profit @Drew Paroski
- lamp.baidu.com


Q&A