

PHPエクステンションの 開発tips

hnw

第3回闇PHP勉強会 (2012/09/01) 発表資料

自己紹介

- @hnw
- カレーとバグが大好物
- 好きな境界値： $2^{53}+1$
(doubleで表現できない最小の自然数)

今日おはなしすること

- 最近PHPエクステンション書きました
- PHP自体をいじるようなエクステンションは初挑戦でした
- 作ってみて気づいた点をとりとめなく話します

php-timecopの紹介

- php-timecopというエクステンションを作りました
 - <https://github.com/hnw/php-timecop>
 - 現在時刻をずらす関数を提供し、
既存の日付系関数/クラスをオーバーライド
 - 現在時刻に依存する関数の例：`strtotime("+1 month")`


```
timecop-example.php - emacs@caramel.local
<?php
timecop_freeze(0);

/* 1970-01-01 09:00:00 */
echo date("Y-m-d H:i:s"), "<br>\n";

/* 2000-01-01 */
$dt = new DateTime();
echo $dt->modify("+30 year")->format("Y-m-d"), "<br>\n";

---:--- timecop-example.php All L1 (PHP JaSp Flymake A
Mark set
```

php-timecop動作例

Cソース上の関数の分類

- PHP_FUNCTION
 - PHPの関数として実行可能
- PHPAPI
 - 他のエクステンションから呼び出せる
- 通常関数
 - そのエクステンション内で閉じている

Cソース上の関数の分類

- 今回は既存の日付系関数のラッパー関数を作った
 - dateエクステンションのC関数を呼びたい！
 - 呼びたい関数がPHPAPIになってない…
- PHP_FUNCTIONを呼ぶことにした

エクステンション内から PHP関数を呼び出す

- zend_call_functionを利用した

```
int zend_call_function(zend_fcall_info *fci,  
                      zend_fcall_info_cache *fci_cache TSRMLS_DC)
```


3重ポインタで悩んだ

- zend_fcall_infoのparamsがなぜzval***なのか？
- コピーオンライトを実現するためだと想像

```
zend_API.h - emacs@caramel.local
typedef struct _zend_fcall_info {
    size_t size;
    HashTable *function_table;
    zval *function_name;
    HashTable *symbol_table;
    zval **retval_ptr_ptr;
    zend_uint param_count;
    zval ***params;
    zval *object_ptr;
    zend_bool no_separation;
} zend_fcall_info;
```

---:--- zend_API.h 5% L49 (C/l Abbrev)

既存のクラスを継承する

- zend_register_internal_class_exの第2引数で親クラスを指定する

```
timecop.c - emacs@caramel.local — (67 x 16)
if (zend_hash_find(CG(class_table),
                    "datetime", sizeof("datetime"),
                    (void **) &pce) == FAILURE) {
    return SUCCESS; /* DateTime must be initialized before */
}
parent_ce = *pce;

INIT_CLASS_ENTRY(ce, "TimecopDateTime",
                 timecop_datetime_class_functions);
self_ce = zend_register_internal_class_ex(&ce, parent_ce, NULL
TSRMLS_CC);
self_ce->create_object = parent_ce->create_object;

---:--- timecop.c      34% L308  Git:master  (C/l Abbrev)
```


関数のオーバーライド

- EG(function_table)をzend_hash_updateする
 - mbstringエクステンションからコピペしました
- クラスについてはEG(class_table)を書き換え

PHPバージョン間の違い

- 当初PHP5.4で開発していた
- PHP5.2でビルドしたら怒られた
 - PHP_FE_END (5.3で定義された)
 - zend_fcall_info_init (5.3から引数の個数が変わった)
 - zend_parse_parameters (“*”への対応が5.3から)

→5.2対応する気なら最初から5.2で開発すべき

php-build+phpenv便利

- php-build: PHP5.2から最新版までビルドするツール
- phpenv: PHPのバージョンを切り替えるツール
 - phpizeも切り替わる！
 - エクステンション開発にオススメ

Travis CI便利

- PHP 5.2、5.3、5.4を提供
 - 各バージョンについて
「phpize && ./configure && make test」 できる
- pushするだけで3バージョンのmake testが自動実行
 - .travis.yml参照
- 緑色になっていると自己満足できる

php-timecop

build status **passing**

DESCRIPTION

make testのオプション

- NO_INTERACTION=1
 - テスト結果をメールするかどうか聞かない
 - Travis CIでテストするときは必須
- REPORT_EXIT_STATUS=1
 - make testがしくじったらexit code非ゼロを返す
 - いまは動いてない。GitHub上でissueになっている。

疑問点など

- ZTSなにそれうまいの？
 - 対応できてない気がする
 - しかし実運用してる人っているんですかね
- `mbstring.func_overload`って正しくは`override`じゃね？
 - あの方が間違うとは思えないし、どうなんでしょね

参考にした資料

- 「[PHP Extension Writing](#)」（2009年、PDF109ページ）
 - マクロの説明が超親切に書いてあってオススメ
- PHPソースコード

ご静聴
ありがとうございます
ございました