

Welcome!

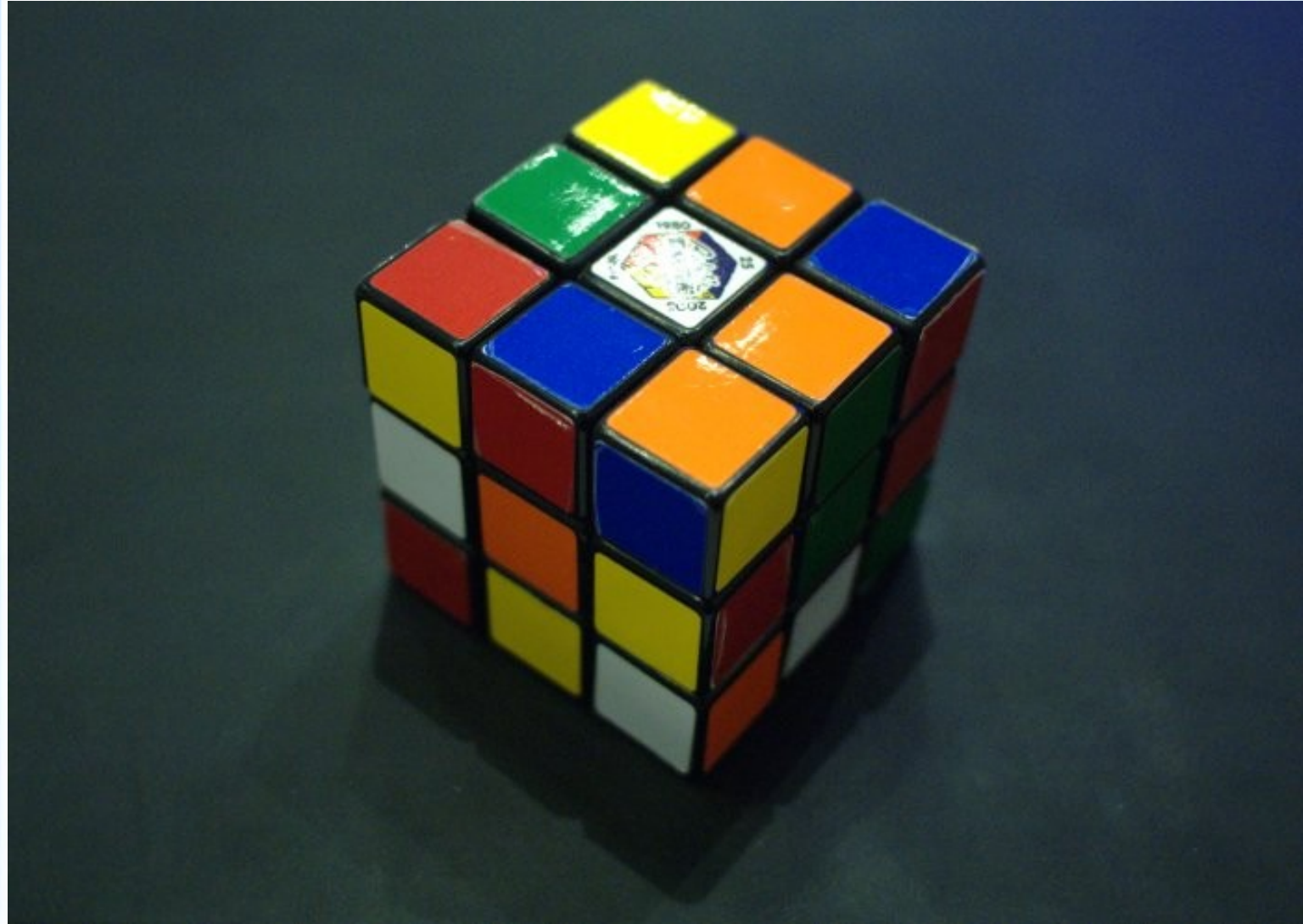
Share your information



I Do Not Need a Debugger

- `printf()`, `var_dump()` and `echo` are good enough

I Do Not Need a Debugger



I Do Need a Debugger

- Xdebug: An Open Source debugging tool
- About 4 years old
- Version 2 is about to be released
- Works (atleast) on Linux and Mac and Windows

Installing Xdebug

With PEAR:

- `pear install xdebug-beta`

Without PEAR:

- `wget http://xdebug.org/files/xdebug-2.0.0beta5.tgz`
- `tar -xvzf xdebug-2.0.0beta5.tgz`
- `cd xdebug-2.0.0beta5`
- `phpize`
- `./configure`
- `make`
- `make install`

Installing Xdebug

- in php.ini add: zend_extension=/full/path/to/xdebug.so
- Use zend_extension_debug for debug builds of PHP

Installing Xdebug

- Download the .dll for your PHP version from <http://pecl4win.php.net/>
- in php.ini add: `zend_extension_ts=c:\php\xdebug.dll`

Installing Xdebug

- `--enable-versioning` prevent Xdebug from loading
- Zend's extensions (optimizer, debugger, cache) prohibit Xdebug (and other non-Zend zend-extensions) from loading

- Stack overflow in PHP
- Infinite recursion

Xdebug protects with:

```
xdebug.max_nesting_level=4
```

Fatal error: Maximum function nesting level of '4' reached, aborting!
in **/home/httpd/html/test/xdebug/infinite.php** on line **3**

Call Stack

| # | Function | Location |
|---|----------|---|
| 1 | {main}() | /home/httpd/html/test/xdebug/infinite.php:0 |
| 2 | a() | /home/httpd/html/test/xdebug/infinite.php:6 |
| 3 | a() | /home/httpd/html/test/xdebug/infinite.php:3 |
| 4 | a() | /home/httpd/html/test/xdebug/infinite.php:3 |

Help By Error Messages

Warning: fopen(ezctest) [[function fopen](#)]: failed to open stream: No such file or directory in /home/httpd/html/test/ezc/dr_mail_test.php on line 147

... is kinda useless.

Warning: fopen(ezctest) [[function fopen](#)]: failed to open stream: No such file or directory in /home/httpd/html/test/ezc/dr_mail_test.php on line 146

| Call Stack | | |
|------------|--|--|
| # | Function | Location |
| 1 | {main}() | /home/httpd/html/test/ezc/dr_mail_test.php:0 |
| 2 | ezcSingleFileSet->__construct('ezctest' (length=11)) | /home/httpd/html/test/ezc/dr_mail_test.php:184 |
| 3 | fopen('ezctest' (length=11) , 'r' (length=1)) | /home/httpd/html/test/ezc/dr_mail_test.php:146 |

| Variables in local scope (#2) | |
|-------------------------------|-----------------------|
| Variable | Value |
| \$file = | 'ezctest' (length=11) |

xdebug.collect_params=1 and xdebug.show_local_vars=1

Debugging With Request Variables

Warning: *fopen(ezctest) [function.fopen]: failed to open stream: No such file or directory in /home/httpd/html/test/ezc/dr_mail_test.php on line 149*

Call Stack

| # | Function | Location |
|--------------------|---------------------------------|--|
| 1 | {main}() | /home/httpd/html/test/ezc/dr_mail_test.php:0 |
| 2 | ezcSingleFileSet->__construct() | /home/httpd/html/test/ezc/dr_mail_test.php:187 |
| 3 | fopen () | /home/httpd/html/test/ezc/dr_mail_test.php:149 |
| Dump \$_GET | | |
| | \$_GET['user'] = | 'derick' (length=6) |
| | \$_GET['password'] = | 'hemmelig' (length=8) |

xdebug.dump.GET=*

xdebug.dump.POST=username,password

Also for: COOKIE, ENV, FILES, REQUEST, SERVER and SESSION

Pretty Printing Variables

```
array
  0 => null
  1 => 3.1415926
  2 =>
    array
      'dutch' =>
        object(locale)[1]
          public 'lang' => 'nl'
          public 'variation' => 'NL'
          public 'charsets' =>
            array
              0 => 'iso-8859-1'
              1 => 'iso-8859-15'
          private 'mb_supported' => false
          private 'self' =>
            &object(locale)[1]
          protected 'id' => 1
```

<?php

xdebug_var_dump(); // or just var_dump()

?>

How Many Functions

```
<?php  
echo xdebug_get_function_count();  
?>
```

```
<?php  
echo xdebug_get_function_count();  
?>
```

How Much Time Does It Take

```
<pre>
<?php
ini_set('include_path', '/home/httpd/ezcomponents/trunk');
require_once 'Base/src/base.php';

function __autoload( $className )
{
    ezcBase::autoload( $className );
}

$config = ezcConfigurationManager::getInstance();
$config->init( 'ezcConfigurationIniReader', dirname( __FILE__ ) . '/examples' );

echo "Time Index: ", xdebug_time_index(), "\n";
$password = $config->getSetting( 'settings', 'db', 'password' );
echo "The password is <$password>.\n";
echo "Time Index: ", xdebug_time_index(), "\n";
?>
```

How Much Memory Does It Use

```
<pre>
<?php
require 'ezc_mail_setup.php';
error_reporting(E_ALL);

$parser = new ezcMailParser();
$set = new ezcSingleFileSet( dirname(__FILE__).'/ezcmailtest.mail' );
echo "Memory: ", xdebug_memory_usage(), " bytes\n\n";

$mail = $parser->parseMail( $set );
foreach( $mail as $mailPart )
{
    echo "From: {$mailPart->from->email}\n";
    echo "Subject: {$mailPart->subject}\n";
}
unset( $mail );

echo "\nMaximum Memory: ", xdebug_peak_memory_usage(), " bytes\n";
?>
```

```

mc - kossu.home.derickrethans.nl:~/j pilot/Backup - [derick@kossu]
27.6247 17664184 -> ezTemplateExpressionSourceToTstParser->__construct
27.6248 17664184 -> ezTemplateSourceToTstParser->__construct() /Temp
27.6250 17664248 -> ezTemplateSourceToTstParser->parseRequiredType() /
27.6251 17664248 -> is_string() /Template/src/parsers/source_to_tst/i
27.6252 17664248 -> is_object() /Template/src/parsers/source_to_tst/i
27.6252 17664248 -> ezTemplateSourceToTstParser->parse() /Template/s
27.6253 17664248 -> ezTemplateExpressionSourceToTstParser->parseCu
27.6256 17664248 -> ezTemplateSourceToTstParser->findNextElement
27.6256 17664248 -> ezTemplateCursor->atEnd() /Template/src/pa
27.6257 17664248 -> strlen() /Template/src/cursor.php:87
27.6258 17664248 -> ezTemplateSourceToTstParser->skipWhitespac
27.6258 17664248 -> ezTemplateCursor->atEnd() /Template/src/
27.6259 17664248 -> strlen() /Template/src/cursor.php:87
27.6260 17664248 -> ezTemplateCursor->pregMatch() /Template/
27.6260 17664248 -> strlen() /Template/src/cursor.php:340
27.6261 17664248 -> substr() /Template/src/cursor.php:343
27.6262 17664760 -> preg_match() /Template/src/cursor.php:3
27.6263 17664392 -> strlen() /Template/src/parsers/source_to
/tmp/trace.2310770352.xt 1,4 All

```

Available settings:

```

xdebug.auto_trace=1
xdebug.trace_output_dir=/tmp
xdebug.collect_params=1
xdebug.collect_return=1
xdebug.trace_options=1

```


demo

Share your information

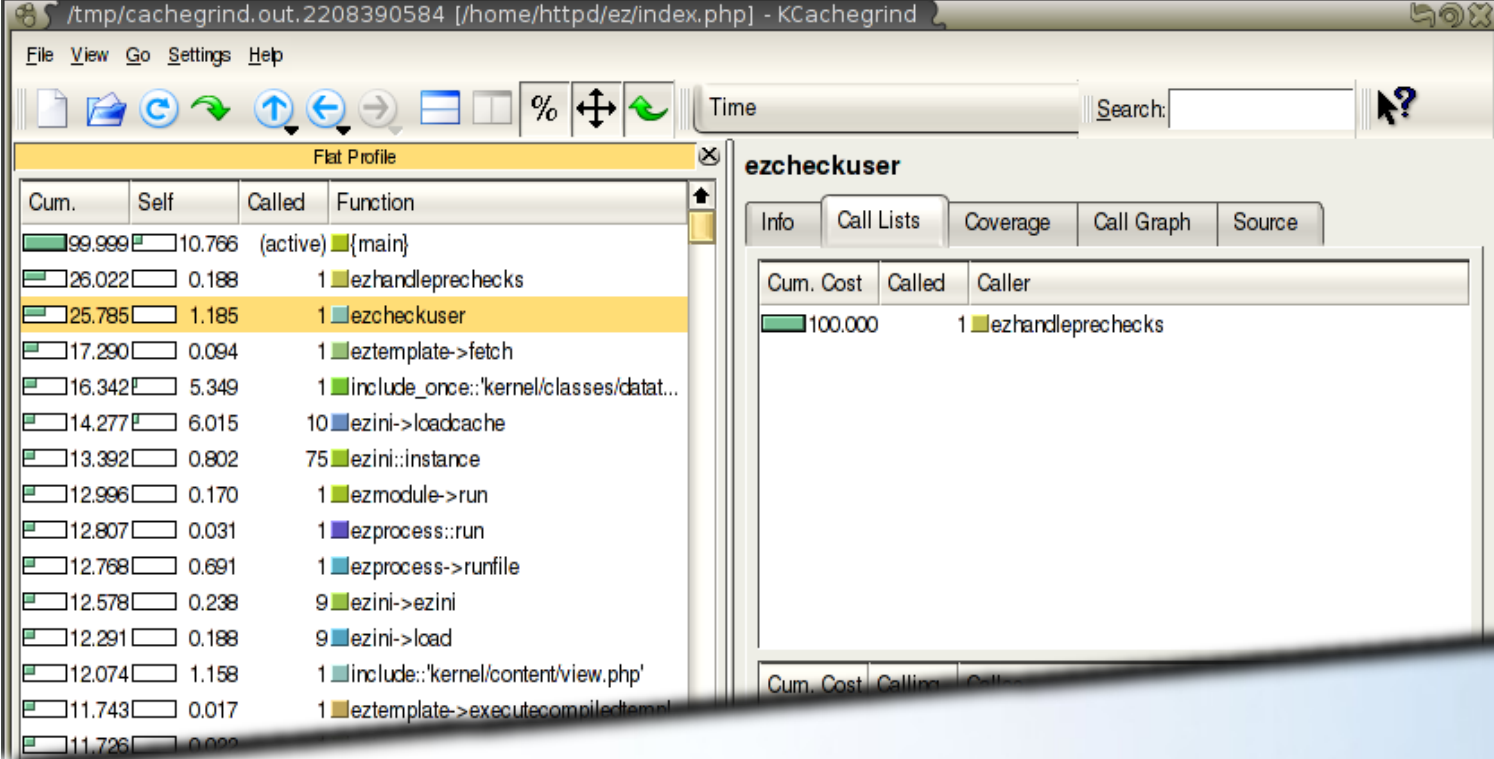
What Code Do I Use

```
34 public function __construct()  
35 {  
36     $args = func_get_args();  
37     parent::__construct( array() );  
38     foreach ( $args as $part )  
39     {  
40         if ( $part instanceof ezcMailPart )  
41         {  
42             $this->parts[] = $part;  
43         }  
44         elseif( is_array( $part ) ) // add each  
45         {  
46             foreach ( $part as $array_part )  
47             {
```

Available functions:

```
xdebug_start_code_coverage();  
xdebug_get_code_coverage();  
xdebug_stop_code_coverage();
```

<http://kossu/coverage/index.htm>



The screenshot displays the KCachegrind web interface. The left pane, 'Flat Profile', shows the following data:

| Cum. | Self | Called | Function |
|--------|--------|----------|--|
| 99.999 | 10.766 | (active) | {main} |
| 26.022 | 0.188 | 1 | ezhandleprechecks |
| 25.785 | 1.185 | 1 | ezcheckuser |
| 17.290 | 0.094 | 1 | eztemplate->fetch |
| 16.342 | 5.349 | 1 | include_once::'kernel/classes/datat... |
| 14.277 | 6.015 | 10 | ezini->loadcache |
| 13.392 | 0.802 | 75 | ezini::instance |
| 12.996 | 0.170 | 1 | ezmodule->run |
| 12.807 | 0.031 | 1 | ezprocess::run |
| 12.768 | 0.691 | 1 | ezprocess->runfile |
| 12.578 | 0.238 | 9 | ezini->ezini |
| 12.291 | 0.188 | 9 | ezini->load |
| 12.074 | 1.158 | 1 | include::'kernel/content/view.php' |
| 11.743 | 0.017 | 1 | eztemplate->executecompiledtempl |
| 11.726 | 0.022 | 1 | eztemplate->executecompiledtempl |

The right pane, 'ezcheckuser', shows the following call list:

| Cum. Cost | Called | Caller |
|-----------|--------|-------------------|
| 100.000 | 1 | ezhandleprechecks |

```
xdebug.profiler_enable=1          ; enable profiler
xdebug.profile_output_dir=/tmp    ; output directory
xdebug.profile_output_name=crc32 ; file extension
```

demo

Share your information

Analyzing Running Scripts

- DBGp, common Debugging protocol
- Cross-language: PHP, Python, Perl...
- Supported in Xdebug 2
- Clients: Komodo, Maguma Workbench, TruStudio (PHP Eclipse Plugin) and many other smaller ones

Let's demo this...

demo

Share your information



Xdebug site: <http://xdebug.org>

Xdebug documentation: <http://xdebug.org/docs.php>

DBGp specification: <http://xdebug.org/docs-dbgp.php>