

Mihai Brehar

<http://www.mihaibrehar.ro/blog/>

**If you write code,
you're writing bugs —
it's a simple fact of
life (Marco Tabini)**

The 6 stages of debugging:

1)That can't happen

2)That doesn't happen on my machine

3)That shouldn't happen

4)Why does that happen?

5)Oh, I see

6)How did that ever work?

Fatal error: Call to undefined method Dog::sfsd() in /home/mihai/public_html/xdebug/class.Dog.php on line 15

echo \$var;

die(„checkpoint1”);

print_r(\$array);

\$var_dump(\$array);

error_log(„checkpoint2”);

debug_print_backtrace();

Xdebug

<http://xdebug.org/>

APD

<http://www.php.net/manual/en/book.apd.php>

DBG

<http://www.php-debugger.com/dbg/>

Zend Studio

<http://www.zend.com/en/products/studio/>

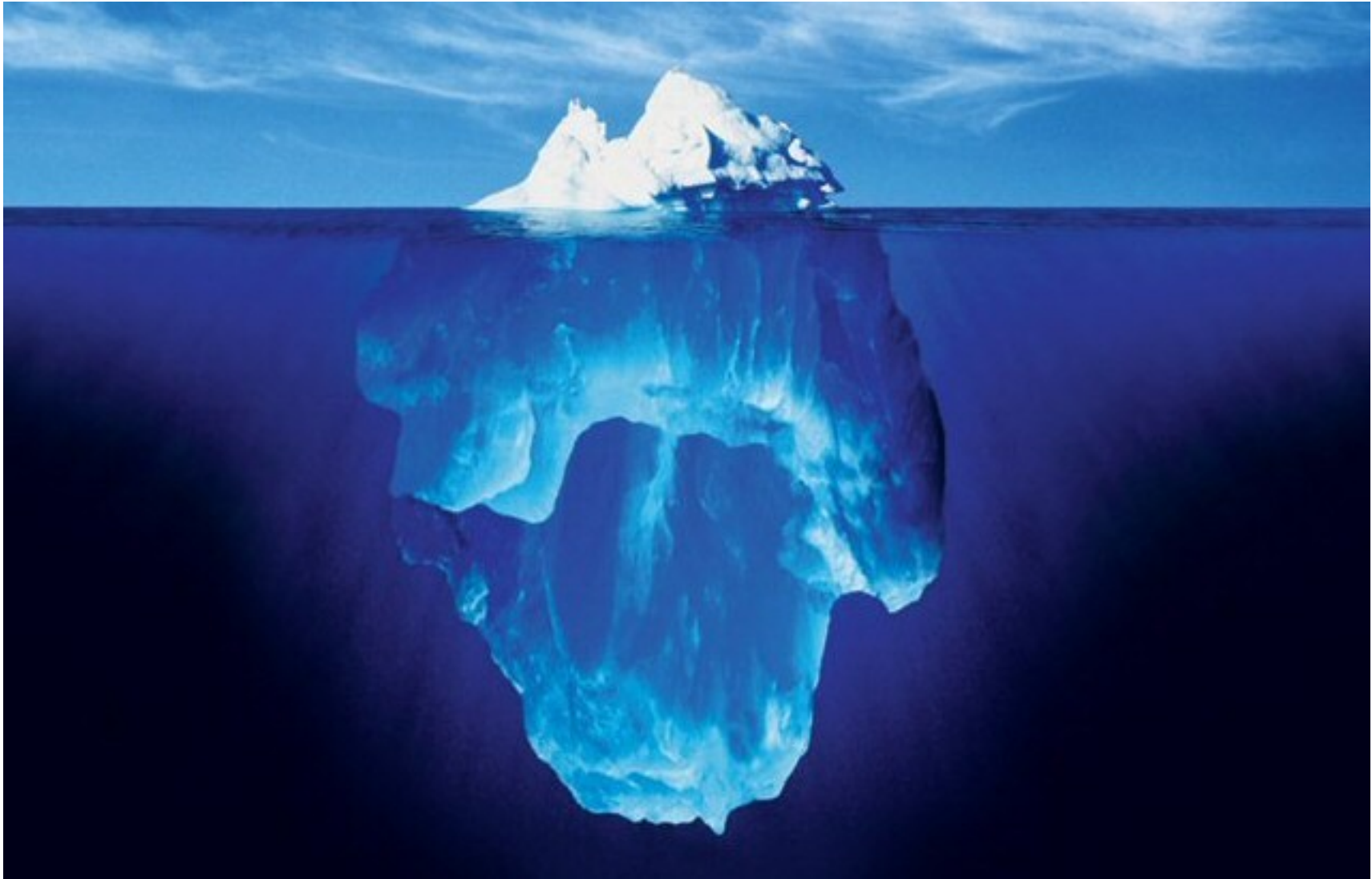


Lansat acum ~6 ani
Autor: Derick Rethans

Ce oferă Xdebug?

- **stack traces**
- **profiling**
- **code coverage analysis**
- **interactive debugging**
- **protectie pentru recursivitate
infinita**

Fatal error: Call to undefined method Dog::sfsd() in /home/mihai/public_html/xdebug/class.Dog.php on line 15



Stack trace

(!) Fatal error: Call to undefined method Dog::asdf() in /home/mihai/public_html/xdebug/class.Dog.php on line 17

Call Stack

| # | Time | Memory | Function | Location |
|---|--------|--------|------------------|--------------------|
| 1 | 0.0006 | 100448 | {main}() | ../xdebug.php:0 |
| 2 | 0.0018 | 123144 | Dog->makeSound() | ../xdebug.php:13 |
| 3 | 0.0018 | 123144 | Dog->bark() | ../class.Dog.php:6 |

xdebug.collect_params=4

(!) Fatal error: Call to undefined method Dog::asdf() in /home/mihai/public_html/xdebug/class.Dog.php on line 17

Call Stack

| # | Time | Memory | Function | Location |
|---|--------|--------|--|--------------------|
| 1 | 0.0007 | 100448 | {main}() | ../xdebug.php:0 |
| 2 | 0.0018 | 123656 | Dog->makeSound() | ../xdebug.php:13 |
| 3 | 0.0018 | 123728 | Dog->bark(\$numTimes = 3, \$abc = 'def') | ../class.Dog.php:6 |

xdebug.dump.GET = *

(!) Fatal error: Call to undefined method Dog::asdf() in /home/mihai/public_html/xdebug/class.Dog.php on line 17

Call Stack

| # | Time | Memory | Function | Location |
|---|--------|--------|--|--------------------|
| 1 | 0.0009 | 100696 | {main}() | ../xdebug.php:0 |
| 2 | 0.0022 | 123904 | Dog->makeSound() | ../xdebug.php:13 |
| 3 | 0.0033 | 123976 | Dog->bark(\$numTimes = 3, \$abc = 'def') | ../class.Dog.php:6 |

Dump \$_GET

```
$_GET['dog_id'] = string '7' (length=1)
$_GET['cat_id'] = string '32' (length=2)
```

xdebug.show_local_vars = 1

(!) Fatal error: Call to undefined method Dog::asdf() in /home/mihai/public_html/xdebug/class.Dog.php on line 17

Call Stack

| # | Time | Memory | Function | Location |
|---|--------|--------|--|--------------------|
| 1 | 0.0007 | 99344 | {main}() | ../xdebug.php:0 |
| 2 | 0.0019 | 122552 | Dog->makeSound() | ../xdebug.php:13 |
| 3 | 0.0019 | 122624 | Dog->bark(\$numTimes = 3, \$abc = 'def') | ../class.Dog.php:6 |

Variables in local scope (#3)

| |
|-----------------------------------|
| \$abc = string 'def' (length=3) |
| \$i = int 3 |
| \$test = string 'test' (length=4) |
| \$numTimes = int 3 |

Overload var_dump

array

```
'SCRIPT_URL' => string '/~mihai/xdebug/index.php' (length=24)
'SCRIPT_URI' => string 'http://127.0.0.1/~mihai/xdebug/index.php' (length=40)
'HTTP_HOST' => string '127.0.0.1' (length=9)
'QUERY_STRING' => string '' (length=0)
'REQUEST_URI' => string '/~mihai/xdebug/index.php' (length=24)
'SCRIPT_NAME' => string '/~mihai/xdebug/index.php' (length=24)
'PHP_SELF' => string '/~mihai/xdebug/index.php' (length=24)
'REQUEST_TIME' => int 1240613525
'argv' =>
  array
    empty
'argc' => int 0
```

Profiler

xdebug.profiler_enable=1

xdebug.profiler_output_dir=/tmp

The screenshot shows the Xdebug Profiler web interface. The browser address bar indicates the URL `cachegrind.out.6977 [/home/mihai/public_html/xdebug/index.php]`. The interface includes a menu bar (File, View, Go, Settings, Help) and a toolbar with buttons for Open, Reload, Force Dump, Up, Back, Forward, and a percentage input field for 'Show Absolute Costs'. A dropdown menu is set to 'Time'.

The main content area is divided into two panels:

- Flat Profile:** A table showing the execution time for each function call. The top row is highlighted in blue.
- Call Graph:** A tree view showing the sequence of function calls. The top row is highlighted in blue.

At the bottom, the status bar displays: `cachegrind.out.6977 [1] - Total Time Cost: 1 366`

| Incl. | Self | Called Function | Location |
|--------|-------|-----------------------------|---------------------|
| 100.15 | 85.51 | (0) {main} | index.php |
| 9.00 | 3.00 | 1 Dog->makeSound | class.Dog.php |
| 5.93 | 5.93 | 1 Dog->bark | class.Dog.php |
| 2.49 | 2.20 | 1 Cat->makeSound | class.Cat.php |
| 2.12 | 2.12 | 2 AnimalFactory::factory | class.AnimalFactory |
| 0.51 | 0.51 | 1 require_once::/home/mi... | class.Animal.php |
| 0.22 | 0.22 | 1 Cat->meow | class.Cat.php |
| 0.22 | 0.22 | 1 require_once::/home/mi... | class.AnimalFactory |
| 0.22 | 0.22 | 1 require_once::/home/mi... | class.Cat.php |
| 0.07 | 0.07 | 1 require_once::/home/mi... | class.Dog.php |

| Incl. | Self | Distance | Calling | Callee |
|-------|------|----------|--|--------|
| 8.99 | 3.00 | 1 | 1 Dog->makeSound (class.Dog.php) | |
| 5.92 | 5.92 | 2 | 1 Dog->bark (class.Dog.php) | |
| 2.49 | 2.19 | 1 | 1 Cat->makeSound (class.Cat.php) | |
| 2.12 | 2.12 | 1 | 2 AnimalFactory::factory (class Ani... | |
| 0.51 | 0.51 | 1 | 1 require_once::/home/mihai/public... | |
| 0.22 | 0.22 | 2 | 1 Cat->meow (class.Cat.php) | |
| 0.22 | 0.22 | 1 | 1 require once::/home/mihai/public... | |

Integrarea cu Eclipse

Q: Does God control everything that happens in my life?

A: He could, if he used the debugger, but it's tedious to step through all those variables.

Name: xdebug test

Server Advanced Common

Server

Server Debugger: XDebug

PHP Server: Default PHP Web Server New Configure...

Test Debugger

File

/xdebug/xdebug.php Browse

Breakpoint

Break at First Line

URL

Auto Generate

URL: http://localhost/ /~mihai/xdebug/xdebug.php

Apply Revert

Debug Close



Debug

xdebug test [PHP Web Page]

- PHP Application
 - Remote Launch (stepping)
 - class.Dog.php.Dog->bark : lineno 11
 - class.Dog.php.Dog->makeSound : line
 - xdebug.php.{main} : lineno 13

Variables Breakpoints Expressions Debug Outp

```
$_$ "i" = 1  
$_$ "$numTimes" = 3
```

```
class.Dog.php  
1 <?php  
2  
3 class Dog extends Animal{  
4  
5     public function makeSound(){  
6         $this->bark(3, "def");  
7     }  
8  
9     private function bark($numTimes, $abc){  
10        $test = "test";  
11        for ($i = 0; $i < $numTimes; $i++){  
12            echo "ham ";  
13        }  
14  
15        echo "<br />";  
16  
17        $this->asdf();  
18    }  
19 }
```

Mulțumesc!

întrebări?

--

Mihai Brehar

<http://www.mihaibrehar.ro/blog/>