Xdebug

Forum PHP - Paris, France - Nov 10, 2010

Derick Rethans - derick@derickrethans.nl - twitter: @derickr

http://derickrethans.nl/talks.html

http://joind.in/2091

About Me

Derick Rethans

- Dutchman living in London
- PHP development
- Author of the mcrypt, input_filter, dbus, translit and date/time extensions
- Author of Xdebug
- Contributor to the Apache Zeta Components Incubator project (formerly eZ Components)
- Freelancer doing PHP (internals) development

I Do Need a Debugger

- Xdebug: An Open Source debugging tool
- About 8 years old
- Works on "every" operating system
- Version 2.1 released earlier this year
- PHP 5.1, 5.2, 5.3 and trunk

Installing Xdebug

Tailored installation instructions:

Help By Error Messages

```
Warning: DOMDocument::load() [domdocument.load]: Document is empty in
/home/httpd/presentations,
line: 1 in /home/httpd/pres2/show2.php on line 165
```

```
|( ! ) Warning: DOMDocument::load() [domdocument.load]: Document is empty in /home/httpd/|
|presentations,_line:_1_in_/home/httpd/pres2/show2.php_on_line_165_____________________|
|Call_Stack_____________________________________________________________________________|
|#|Time__|Memory_|Function_____________|Location_____________________________________|
|1|0.0022|_786328|{main}(_)____________|../show2.php:0_______________________________|
|2|0.0180|1397248|Presentation->display(_)|../show2.php:114_____________________________|
|3|0.0183|1399032|DOMDocument->load(_)____|../show2.php:165_____________________________|
```

xdebug.default_enable=1
html_errors=1

```
|( ! ) Warning: DOMDocument::load() [domdocument.load]: Document is empty in /home/httpd/|
|presentations,_line:_1_in_/home/httpd/pres2/show2.php_on_line_165_____________________|
|Call_Stack_____________________________________________________________________________|
|#|Time__|Memory_|Function_____________________|Location_____________________________|
|1|0.0024|_786328|{main}(_)____________________|../show2.php:0_______________________|
|2|0.0183|1397248|Presentation->display(_null_)__|../show2.php:114_____________________|
|3|0.0185|1399032|DOMDocument->load(_string(32)_)|../show2.php:165_____________________|
```

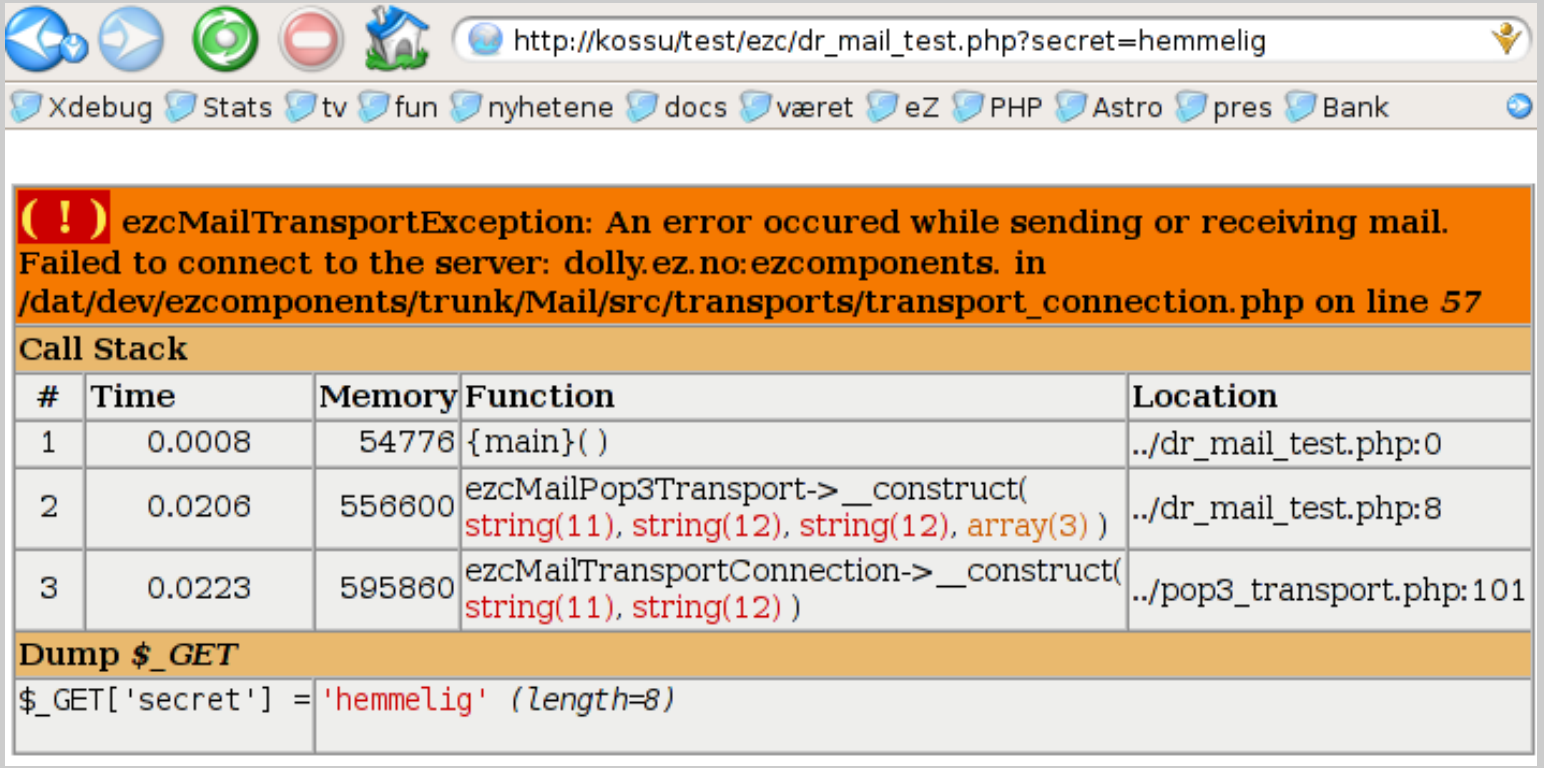
xdebug.collect_params=1

```
|( ! ) Warning: DOMDocument::load() [domdocument.load]: Document is empty in /home/httpd/|
|presentations,_line:_1_in_/home/httpd/pres2/show2.php_on_line_165_____________________|
|Call_Stack_____________________________________________________________________________|
|#|Time__|Memory_|Function_____________________|Location_____________________________|
|1|0.0022|_787720|{main}(_)____________________|../show2.php:0_______________________|
|2|0.0183|1398552|Presentation->display(_null_)__|../show2.php:114_____________________|
|3|0.0186|1400336|DOMDocument->load(_string(32)_)|../show2.php:165_____________________|
```

xdebug.collect_params=2

```
|( ! ) Warning: DOMDocument::load() [domdocument.load]: Document is empty in /home/httpd/          |
|presentations,_line:_1_in_/home/httpd/pres2/show2.php_on_line_165_____________________|
```

Debugging With Request Variables



xdebug.dump.GET=*
xdebug.dump.POST=username,password
Also for: COOKIE, ENV, FILES, REQUEST, SERVER and SESSION

Pretty Printing Variables

```
array(3) { [0]=> NULL [1]=> float(3.141592654) [2]=> array(1) { ["dutch"]=> object(locale)#1 (6) {
["lang"]=> string(2) "nl" ["variation"]=> string(2000)
"abcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcda
bcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabcdabc
dab
```

Scream

- PHP's @ operator hides warnings and errors
- xdebug.scream=1 makes PHP ignore @

Function trace

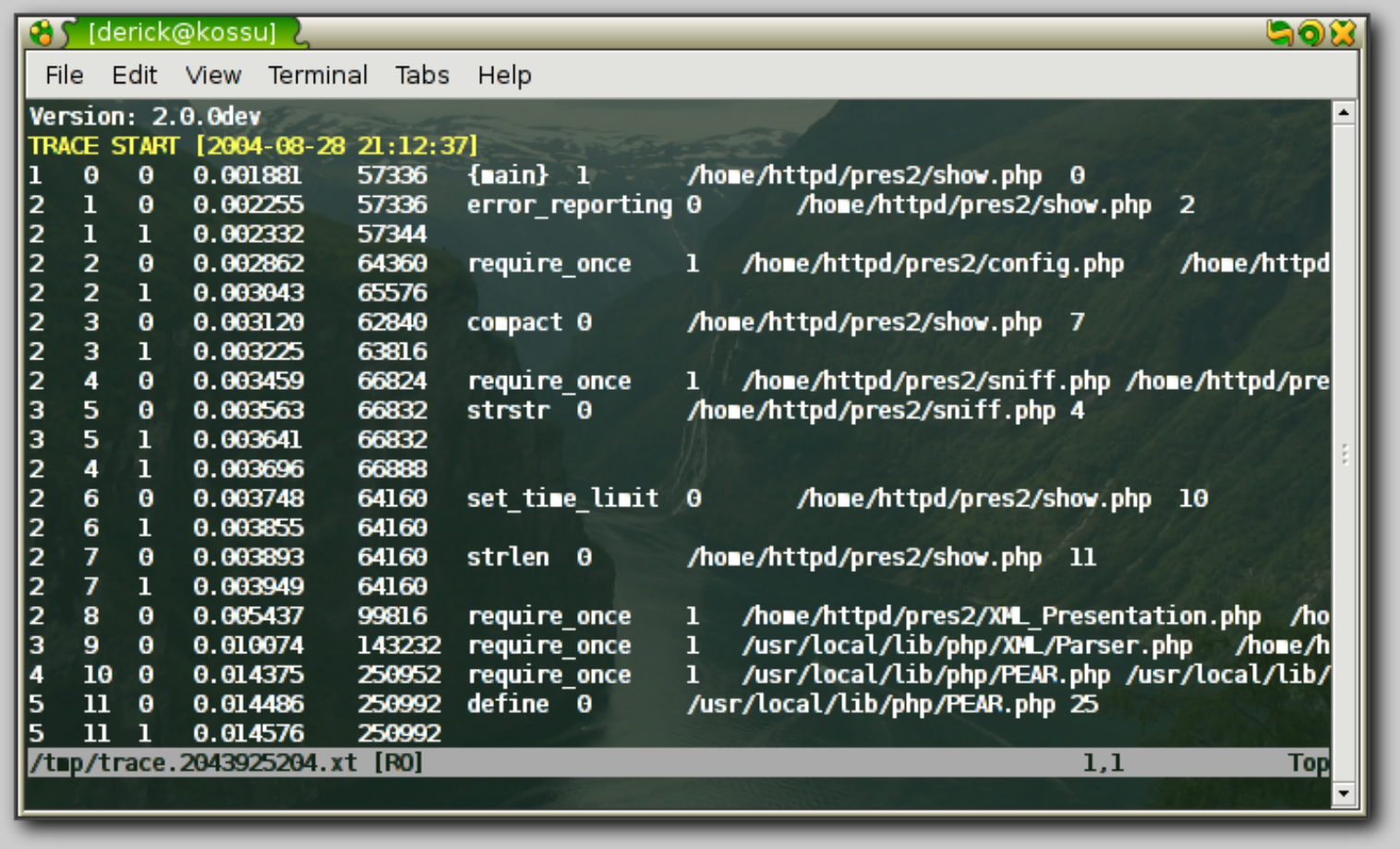
```
TRACE START [2010-03-24 11:03:12]
    0.0022     787776   -> {main}() /home/httpd/pres2/show2.php:0
    0.0028     803160     -> require(/home/derick/dev/ezcomponents/trunk/Base/src/
ezc_bootstrap.php)/home/httpd/pres2/show2.php:2
    0.0029     803552       -> dirname('/home/derick/dev/ezcomponents/trunk/Base/src/
ezc_bootstrap.php')/home/derick/dev/ezcomponents/trunk/Base/src/ezc_bootstrap.php:12
    0.0030     803968       -> explode('/', '/home/derick/dev/ezcomponents/trunk/Base/src')/home/
derick/dev/ezcomponents/trunk/Base/src/ezc_bootstrap.php:13
    0.0031     807352       -> count(array (0 => '', 1 => 'home', 2 => 'derick', 3 => 'dev', 4 =>
'ezcomponents', 5 => 'trunk', 6 => 'Base', 7 => 'src'))/home/derick/dev/ezcomponents/trunk/Base/src/
ezc_bootstrap.php:15
    0.0032     807800       -> array_slice(array (0 => '', 1 => 'home', 2 => 'derick', 3 => 'dev', 4
=> 'ezcomponents', 5 => 'trunk', 6 => 'Base', 7 => 'src'), 0, -2)/home/derick/dev/ezcomponents/
trunk/Base/src/ezc_bootstrap.php:17
    0.0033     808952       -> join('/', array (0 => '', 1 => 'home', 2 => 'derick', 3 => 'dev', 4
=> 'ezcomponents', 5 => 'trunk'))/home/derick/dev/ezcomponents/trunk/Base/src/ezc_bootstrap.php:17
    0.0065    1002544       -> require(/home/derick/dev/ezcomponents/trunk/Base/src/base.php)/home/
derick/dev/ezcomponents/trunk/Base/src/ezc_bootstrap.php:18
    0.0066     990560     -> __autoload('ezcTemplateCustomFunction')/home/derick/dev/ezcomponents/
trunk/Base/src/ezc_bootstrap.php:0
    0.0066     990560       -> ezcBase::autoload('ezcTemplateCustomFunction')/home/derick/dev/
ezcomponents/trunk/Base/src/ezc_bootstrap.php:38
    0.0067     990560         -> ezcBase::setPackageDir()/home/derick/dev/ezcomponents/trunk/Base/
src/base.php:125
    0.0068     990832           -> dirname('/home/derick/dev/ezcomponents/trunk/Base/src/base.php')/
home/derick/dev/ezcomponents/trunk/Base/src/base.php:244
    0.0069     990728         -> array_key_exists('ezcTemplateCustomFunction', array ())/home/
derick/dev/ezcomponents/trunk/Base/src/base.php:128
    0.0069     990728         -> array_key_exists('ezcTemplateCustomFunction', array ())/home/
derick/dev/ezcomponents/trunk/Base/src/base.php:142
    0.0070     991536         -> preg_match('/^([a-z0-9]*)([A-Z][a-z0-9]*)?([A-Z][a-z0-9]*)?/',
'ezcTemplateCustomFunction', NULL)/home/derick/dev/ezcomponents/trunk/Base/src/base.php:157
    0.0072     993304         -> sizeof(array (0 => 'ezcTemplateCustom', 1 => 'ezc', 2 =>
'Template', 3 => 'Custom'))/home/derick/dev/ezcomponents/trunk/Base/src/base.php:161
    0.0073     993664         -> strtolower('Template_Custom_autoload.php')/home/derick/dev/
ezcomponents/trunk/Base/src/base.php:165
    0.0073     993496         -> ezcBase::requireFile('template_custom_autoload.php',
'ezcTemplateCustomFunction', 'ezc')/home/derick/dev/ezcomponents/trunk/Base/src/base.php:167
    0.0074     994184           -> file_exists('/home/derick/dev/ezcomponents/trunk/Base/src/../../
autoload/template_custom_autoload.php')/home/derick/dev/ezcomponents/trunk/Base/src/base.php:278
    0.0075     993848         -> strtolower('Template_autoload.php')/home/derick/dev/ezcomponents/
trunk/Base/src/base.php:175
    0.0076     993888         -> ezcBase::requireFile('template_autoload.php',
'ezcTemplateCustomFunction', 'ezc')/home/derick/dev/ezcomponents/trunk/Base/src/base.php:177
    0.0077     994568           -> file_exists('/home/derick/dev/ezcomponents/trunk/Base/src/../../
autoload/template_autoload.php')/home/derick/dev/ezcomponents/trunk/Base/src/base.php:278
    0.0100    1105728           -> require(/home/derick/dev/ezcomponents/trunk/autoload/
template_autoload.php)/home/derick/dev/ezcomponents/trunk/Base/src/base.php:280
```

Some settings:

```
xdebug.auto_trace=1
xdebug.trace_output_dir=/tmp
xdebug.collect_params=1
xdebug.collect_return=1
xdebug.collect_includes=1
xdebug.collect_assignments=1
```

Function trace to file

Automatic readable format




```
xdebug.auto_trace=1      ; enable tracing
xdebug.trace_format=1    ; selects computerized format
xdebug.trace_options=0   ; sets extra option (1 = append)
```

Function trace

Other functionality

- HTML traces
- Tracing only parts of an application with xdebug_start_trace() and xdebug_stop_trace().
- Fetching the trace file name that is being used with xdebug_get_tracefile_name().
- Changing how much data is shown with xdebug.var_display_max_children, xdebug.var_display_max_data and xdebug.var_display_max_depth.

demo

Simple profiling

- Uses the "computerized" trace files
- A script is provided in the Xdebug source dir: contrib/tracefile-analyser.php:

```
php tracefile-analyser.php /tmp/trace.124683842.xt [sortkey] [element count]
```

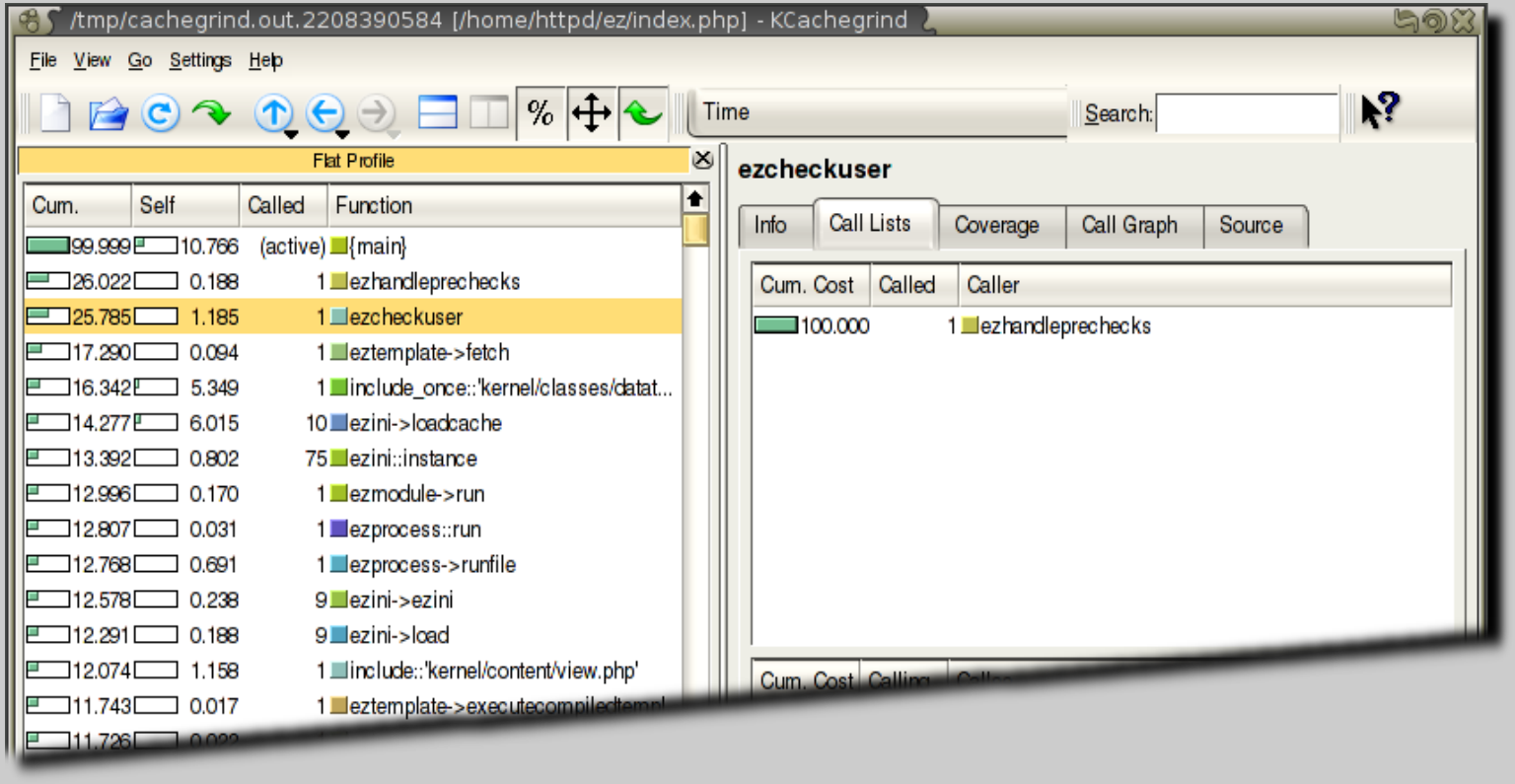
- sort keys: calls, time-inclusive, memory-inclusive, time-own, memory-own

```
Showing the 10 most costly calls sorted by 'time-inclusive'.

                                                    Inclusive        Own
function                                   #calls  time     memory  time     memory
-------------------------------------------------------------------------------------
{main}                                          1  0.0894  1025656  0.0006     5672
Presentation->display                           1  0.0723   408848  0.0004   -11528
ezcTemplate->process                           10  0.0685   243320  0.0005    -2976
ezcTemplateCompiledCode->execute               10  0.0627    14144  0.0009     1672
include                                        10  0.0612    12352  0.0009    -4392
__autoload                                     10  0.0176   781904  0.0006     1080
ezcBase::autoload                              10  0.0170   780824  0.0017   -80728
ezcTemplateCompiledCode::findCompiled          10  0.0094    38776  0.0024    12888
ezcTemplateCompiledCode->checkRequirements     10  0.0094     2968  0.0029     2968
ezcBase::loadFile                              10  0.0093   642320  0.0070   625520
```

Profiling

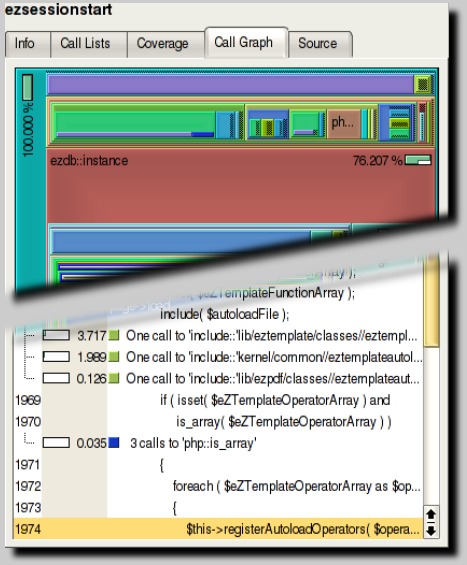
KCacheGrind's Flat Profile and Call List




```
xdebug.profiler_enable=1          ; enable profiler
xdebug.profiler_output_dir=/tmp    ; output directory
xdebug.profiler_output_name=cachegrind.out.%p
```

Profiling

KCacheGrind's Call Graph and Source Annotations

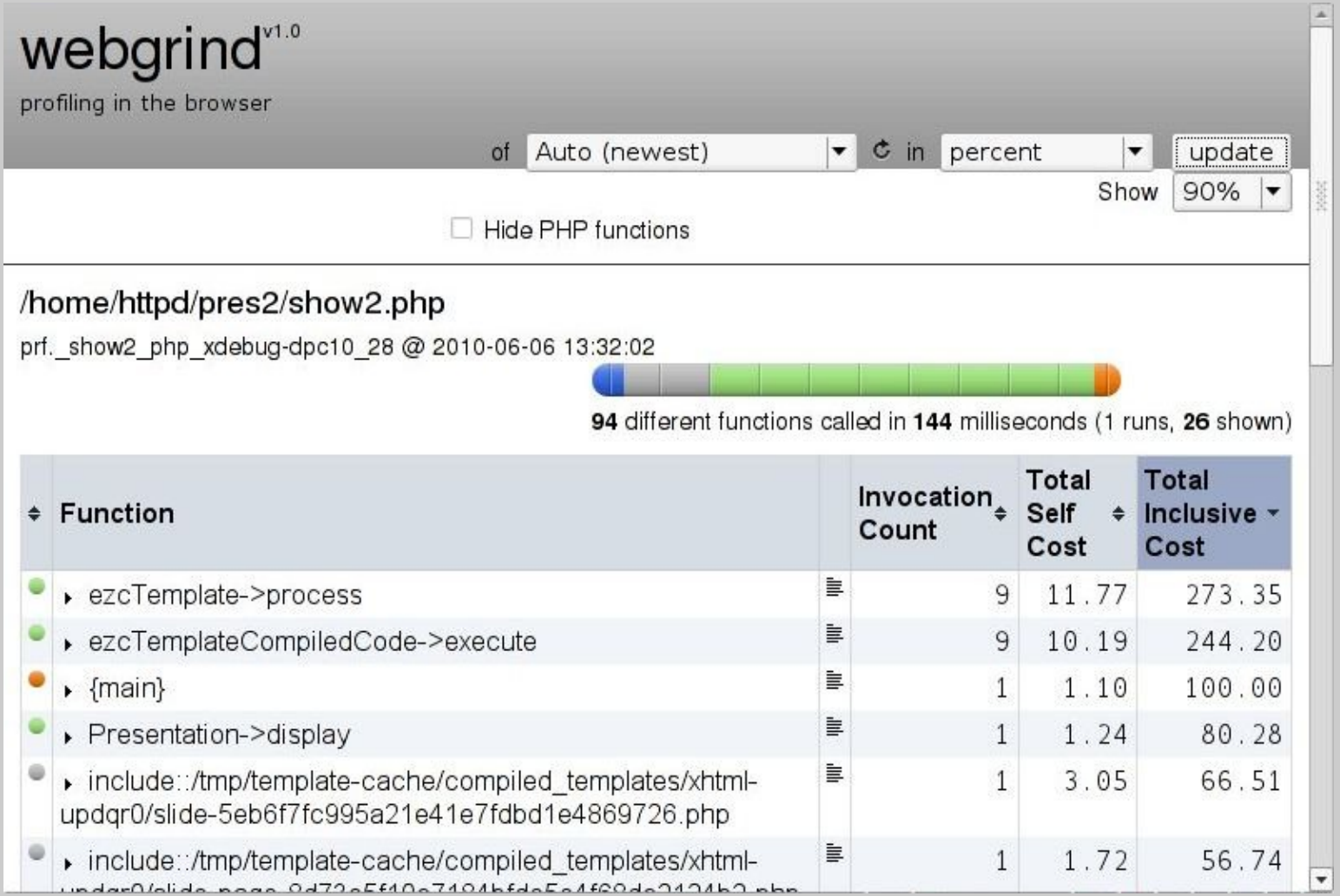


- Call graph
- Area shows time spend
- Stacked to show callees
- Source annotations
- Number of calls
- Total time per function

demo

Webgrind

- Easy to install: extract in a directory in the HTTP root and run



What Code Do I Use

Code Coverage

Functions for obtaining coverage information:

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

Options to xdebug_start_code_coverage(), both only work for files parsed from this moment:

- XDEBUG_CC_UNUSED: Enables scanning of code to figure out which line has executable code.
- XDEBUG_CC_DEAD_CODE: Enables branch analyzation to figure out whether code can be executed.
- http://kossu/coverage/index.html

Analyzing Running Scripts

- DBGp, common Debugging protocol
- Cross-language: PHP, Python, Perl...
- Supported in Xdebug 2

Clients:

- Stand-alone client (Linux (for now)):
- Komodo (Linux, Windows, Mac — commercial)
- Eclipse/PDT (Java based — free)
- Zend Studio for Eclipse (Eclipse-based — commercial)
- Netbeans (Java-based — free)
- PHPStorm (Java-based — commercial)
- MacGDBp (free)
- (and many others:)

Activating the Remote Debugger

php.ini settings:

```
xdebug.remote_enable=1
xdebug.remote_host=localhost
xdebug.remote_port=9000
```

On the shell:

export XDEBUG_CONFIG="idekey=xdebugrocks"

With a browser:

http://pres/show.php?XDEBUG_SESSION_START=xdebugrocks

With browser extensions:

easy Xdebug (FireFox):

Xdebug Helper (Chrome):

Debugging Gotchas

- Xdebug connects to the client
- Remote vs Local debugging
- Debugging proxy: (Komodo-PythonRemoteDebugging)
- xdebug.remote_connect_back

demo

Resources

- Xdebug site: http://xdebug.org
- Xdebug documentation: http://xdebug.org/docs.php
- DBGp specification: http://xdebug.org/docs-dbgp.php
- If you like Xdebug: http://xdebug.org/donate.php
- These slides: http://derickrethans.nl/talks.html