

# Xdebug

Fosdem 2005

February 27th, 2005. Brussels, Belgium

Derick Rethans <[dr@ez.no](mailto:dr@ez.no)>

<http://files.derickrethans.nl/xdebug-fosdem5.pdf>



Share your information

?

```
[derick@localhost derick]$ cat recursive.php
<?php
    function a() {
        a();
    }

    a();
?>
[derick@localhost derick]$ php recursive.php
Segmentation fault
```

**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

```
xdebug.max_nesting_level=4
```

```
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
```

**Warning: `chdir()` [[function.chdir](#)]: No such file or directory (errno 2) in `/dat/docs/magazine/phpa/code/xdebug/listing2.php` on line 34**

### Call Stack

#	Function	Location
1	{main}()	/dat/docs/magazine/phpa/code/xdebug/listing2.php:0
2	change_dir( 'foo123' )	/dat/docs/magazine/phpa/code/xdebug/listing2.php:37
3	<a href="#">chdir</a> ( '/foo123' )	/dat/docs/magazine/phpa/code/xdebug/listing2.php:34

### Variables in local scope (#2)

Variable	Value
\$path =	'/foo123'
\$dir =	'foo123'

```
; enable showing local vars in error messages
xdebug.show_local_vars=1
```

**Warning: chdir(): No such file or directory (errno 2) in /dat/docs/magazine/phpa/code/xdebug/listing2.php on line 36**

### Call Stack

#	Function	Location
1	{main}()	/dat/docs/magazine/phpa/code/xdebug/listing2.php:0
2	change_dir('foo123')	/dat/docs/magazine/phpa/code/xdebug/listing2.php:39
3	chdir('/foo123')	/dat/docs/magazine/phpa/code/xdebug/listing2.php:36

### Dump \$\_GET

`$_GET[42] = 'the answer to life, the universe and everything'`

```

; show all GET variables in error messages
xdebug.dump.GET=*
; show the login and password POST variables
xdebug.dump.POST=login,password
; show the PHP's session cookie
xdebug.dump.COOKIE=PHPSESSID
    
```

Share your information

```

[derick@kossu] - mc - /dat/ez.no-33/var/ezno
TRACE START [2004-06-03 13:52:21]
0.0032 235760 -> {main}() /dat/ez.no-33/index.php:0
0.0036 235840 -> microtime() /dat/ez.no-33/index.php:34
0.0037 235856 -> ob_start() /dat/ez.no-33/index.php:33
0.0037 278880 -> error_reporting() /dat/ez.no-33/index.php:74
0.0084 611512 -> include_once(/dat/ez.no-33/lib/ezutils/classes/ezsys.php) /dat/ez.no-33/index.php:77
0.0106 766520 -> include_once(/dat/ez.no-33/lib/ezutils/classes/ezsys.php) /dat/ez.no-33/lib/ezutils/classes/ezsys.php:
0.0106 766568 -> define('EZ_SYS_DEBUG_INTERNALS', FALSE) /dat/ez.no-33/lib/ezutils/classes/ezsys.php:
0.0107 765856 -> define('EZ_LEVEL_NOTICE', 1) /dat/ez.no-33/lib/ezutils/classes/ezdebug.php:85
0.0107 765872 -> define('EZ_LEVEL_WARNING', 2) /dat/ez.no-33/lib/ezutils/classes/ezdebug.php:86
/tmp/trace.2043925204.xt [R0] 2,1 Top
0.0456 1698456 -> eztextcodec::internalcharset() /dat/ez.no-33/lib/ez18n/classes/eztextcodec.php:581
0.0456 1698456 -> ezcharsetinfo::realcharset('iso-8859-15') /dat/ez.no-33/lib/ez18n/classes/eztextcodec.php:581
0.0457 1698424 -> ezcharsetinfo::aliastable('iso-8859-15') /dat/ez.no-33/lib/ez18n/classes/ezcharsetinfo.php:212
0.0457 1702584 -> is_array(array('ascii' => 'us-ascii', 'latin1' => 'iso-8859-1', 'latin2' => 'is
0.0461 1702664 -> strtolower('iso-8859-15') /dat/ez.no-33/lib/ez18n/classes/ezcharsetinfo.php:213
0.0500 1966816 -> include_once(/dat/ez.no-33/lib/ezlocale/classes/ezlocale.php) /dat/ez.no-33/index.php:24
0.0501 1966904 -> define('EZ_LEVEL_WARNING_INTERNALS', FALSE) /dat/ez.no-33/lib/ezlocale/classes/ezlocale
0.0501 19658 -> ezini::instance() /dat/ez.no-33/index.php:246
0.0502 166224 -> ezini::instance() /dat/ez.no-33/index.php:246
0.0516 1966840 -> ezini::variable('RegionalSettings', 'Debug') /dat/ez.no-33/index.php:247
/tmp/trace.2043925204.xt [R0] 349,5 1%
    
```

```

xdebug.auto_trace=1 ; enable tracing
xdebug.trace_output_dir=/tmp ; trace output directory
xdebug.trace_output_name=crc32 ; tracefile extension
xdebug.collect_params=0 ; parameter collection
xdebug.collect_includes=1 ; collecting includes
xdebug.extended_info=0 ; extended code generation
xdebug.show_mem_delta=1 ; memory difference
    
```

## ~/.vim/filetype.vim

```
augroup filetypedetect
au BufNewFile,BufRead *.xt setf xt
augroup END
```

## ~/.vim/syntax/xt.vim

(<http://xdebug.org/files/xt.vim>)

```
if version < 600
  syntax clear
elseif exists("b:current_syntax")
  finish
endif

syn match begin          "^TRACE START"
syn match end            "^TRACE END"
syn match date           "\[.*\]"

syn match min_memory    "+\d\+"
syn match pls_memory    "-\d\+"
syn match nll_memory    "+0"

syn match level         "->"
syn match lineno        ":\d\+$"

syn match methodcall    "\k\+-->"
syn match staticcall    "\k\+::"
syn match functionb     "\k\+("
syn match functione     ") "

syn match main          "{main}()"
syn match include       "include\(_once\)\(='.\+')

```

## Execution trace to file

*Automatic readable format*

```

[derick@kossu]
File Edit View Terminal Tabs Help
Version: 2.0.0dev
TRACE START [2004-08-28 21:12:37]
1 0 0 0.001881 57336 {main} 1 /home/httpd/pres2/show.php 0
2 1 0 0.002255 57336 error_reporting 0 /home/httpd/pres2/show.php 2
2 1 1 0.002332 57344
2 2 0 0.002862 64360 require_once 1 /home/httpd/pres2/config.php /home/httpd
2 2 1 0.003043 65576
2 3 0 0.003120 62840 compact 0 /home/httpd/pres2/show.php 7
2 3 1 0.003225 63816
2 4 0 0.003459 66824 require_once 1 /home/httpd/pres2/sniff.php /home/httpd/pre
3 5 0 0.003563 66832 strstr 0 /home/httpd/pres2/sniff.php 4
3 5 1 0.003641 66832
2 4 1 0.003696 66888
2 6 0 0.003748 64160 set_time_limit 0 /home/httpd/pres2/show.php 10
2 6 1 0.003855 64160
2 7 0 0.003893 64160 strlen 0 /home/httpd/pres2/show.php 11
2 7 1 0.003949 64160
2 8 0 0.005437 99816 require_once 1 /home/httpd/pres2/XML_Presentation.php /ho
3 9 0 0.010074 143232 require_once 1 /usr/local/lib/php/XML/Parser.php /home/h
4 10 0 0.014375 250952 require_once 1 /usr/local/lib/php/PEAR.php /usr/local/lib/
5 11 0 0.014486 250992 define 0 /usr/local/lib/php/PEAR.php 25
5 11 1 0.014576 250992
/tmp/trace.2043925204.xt [R0] 1,1 Top

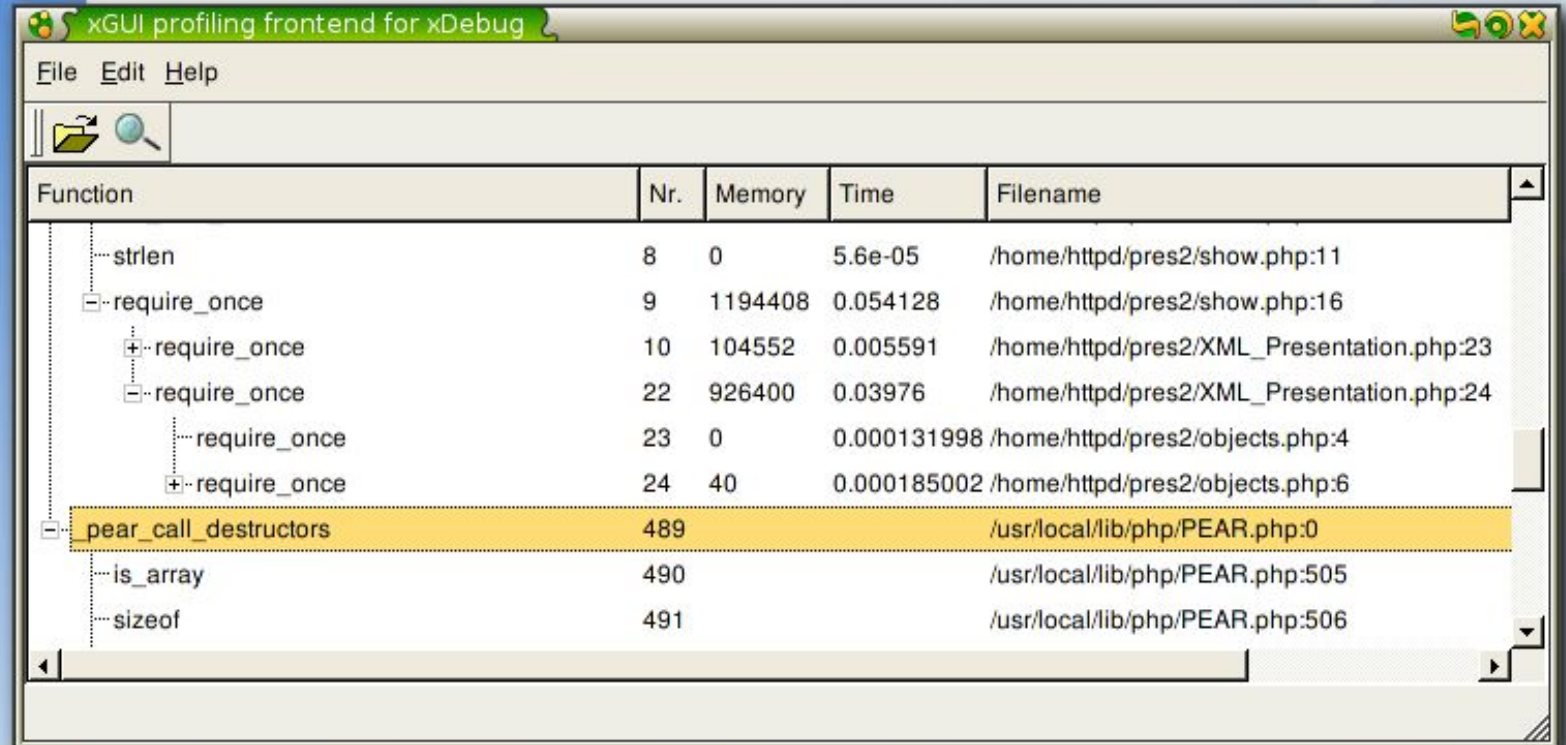
```

```

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

```

Share your information



The screenshot shows a window titled "xGUI profiling frontend for xDebug". It contains a table with the following columns: Function, Nr., Memory, Time, and Filename. The table lists various PHP functions and their call counts, memory usage, and execution times. The function "pear\_call\_destructors" is highlighted in yellow.

Function	Nr.	Memory	Time	Filename
strlen	8	0	5.6e-05	/home/httpd/pres2/show.php:11
require_once	9	1194408	0.054128	/home/httpd/pres2/show.php:16
+ require_once	10	104552	0.005591	/home/httpd/pres2/XML_Presentation.php:23
- require_once	22	926400	0.03976	/home/httpd/pres2/XML_Presentation.php:24
- require_once	23	0	0.000131998	/home/httpd/pres2/objects.php:4
+ require_once	24	40	0.000185002	/home/httpd/pres2/objects.php:6
pear_call_destructors	489			/usr/local/lib/php/PEAR.php:0
is_array	490			/usr/local/lib/php/PEAR.php:505
sizeof	491			/usr/local/lib/php/PEAR.php:506

Still in development, and not yet public.  
Qt, multi-platform (Linux, MacOSX, Windows).

Share your information

The screenshot shows the KCacheGrind application window. The main window title is `/tmp/cachegrind.out.2208390584 [/home/httpd/ez/index.php] - KCacheGrind`. The interface includes a menu bar (File, View, Go, Settings, Help) and a toolbar with various navigation icons. The main content area is split into two panes.

The left pane, titled "Flat Profile", displays a table of functions with their cumulative and self costs. The function `ezcheckuser` is highlighted in yellow.

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	0.020		

The right pane, titled "ezcheckuser", shows a "Call Lists" tab. It displays a table of callers for the selected function.

Cum. Cost	Called	Caller
100.000	1	ezhandleprechecks

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

# Profiling

## KCacheGrind's Call Graph and Source Annotations

Share your information

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

**ezsessionstart**

Info | Call Lists | Coverage | **Call Graph** | Source

ezdb::instance 76.207 %

```

    include( $autoloadFile );
    3.717 One call to 'include::'lib/eztemplate/classes//eztempl...
    1.989 One call to 'include::'kernel/common//eztemplateautol...
    0.126 One call to 'include::'lib/ezpdf/classes//eztemplateaut...
1969     if ( isset( $eZTemplateOperatorArray ) and
1970         is_array( $eZTemplateOperatorArray ) )
    0.035 3 calls to 'php::is_array'
1971     {
1972         foreach ( $eZTemplateOperatorArray as $op...
1973     {
1974         $this->registerAutoloadOperators( $opera...
    
```

## Function to test (bigassfunction.php):

```
<?php
function test_parse_time($f)
{
    if (preg_match("/[0-9]{12}/", $f)) {
        $type = "YYYYMMDDHHii";
    } else if (preg_match("/[0-9]{8}\ [0-9]{4}/", $f)) {
        $type = "YYYYMMDD HHii";
    } else if (preg_match("/[0-9]{4}-[0-9]{2}-[0-9]{2}/", $f)) {
        $type = "YYYY-MM-DD";
    } else {
        $type = "*unknown*";
    }

    echo "This is a $type format.\n";
}
?>
```

## Test driver (basic.t):

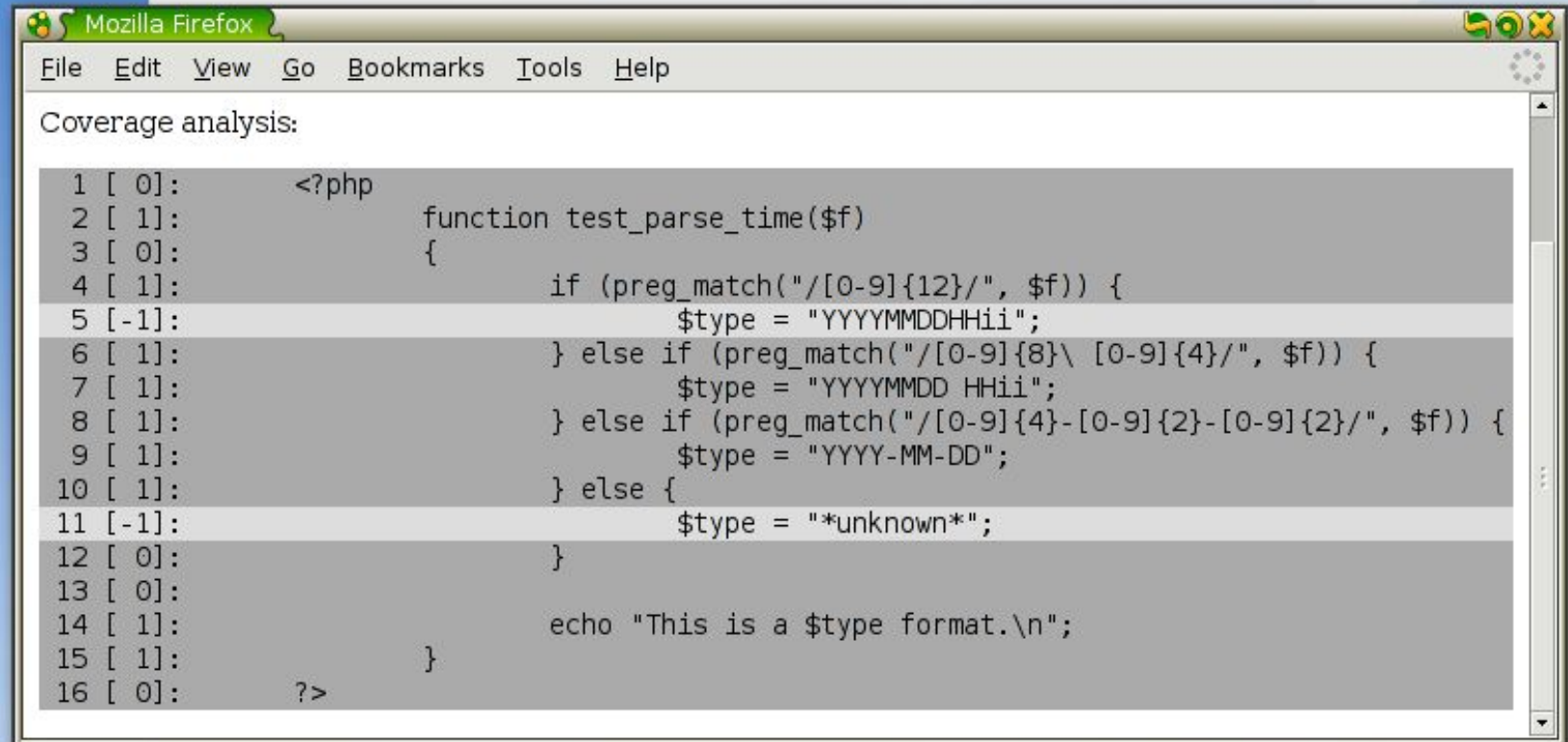
```
<?php
$text = realpath("tests/bigassfunction.php");
include $text;

$formats = array('2004-08-17 21:20', '20040817 2120');
foreach ($formats as $format) {
    test_parse_time($format);
}
?>
```

```
<?php
    echo "Output:\n\n";
    xdebug_start_code_coverage();
    include 'tests/basic.t';
    $cc = xdebug_get_code_coverage();
    xdebug_stop_code_coverage();

    $clean = array();
    $clean[$test] = $cc[$test];

    foreach ($clean as $file => $cc) {
        echo "<pre>";
        $fc = file($file);
        $line_nos = array_keys($cc);
        foreach ($fc as $ln => $line) {
            if (in_array($ln + 1, $line_nos)) {
                $bgc = "aaaaaa";
            } else {
                $bgc = "dddddd";
            }
            echo "<div style='background-color: #\$bgc'>";
            echo @sprintf("%3d [%2d]:\t", $ln + 1, $cc[$ln + 1]);
            echo htmlspecialchars($line);
            echo "</div>";
        }
        echo "</pre>";
    }
?>
```



```
1 [ 0]:      <?php
2 [ 1]:          function test_parse_time($f)
3 [ 0]:          {
4 [ 1]:              if (preg_match("/[0-9]{12}/", $f)) {
5 [-1]:                  $type = "YYYYMMDDHHii";
6 [ 1]:              } else if (preg_match("/[0-9]{8}\ [0-9]{4}/", $f)) {
7 [ 1]:                  $type = "YYYYMMDD HHii";
8 [ 1]:              } else if (preg_match("/[0-9]{4}-[0-9]{2}-[0-9]{2}/", $f)) {
9 [ 1]:                  $type = "YYYY-MM-DD";
10 [ 1]:             } else {
11 [-1]:                 $type = "*unknown*";
12 [ 0]:             }
13 [ 0]:
14 [ 1]:                 echo "This is a $type format.\n";
15 [ 1]:             }
16 [ 0]:      ?>
```

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

- GDB debugger mode
  - GDB like environment
  - Two output possibilities: text and XML
  - Available in Xdebug 1.3.x and 2.x
  
- DBGp debugger mode
  - Language independent
  - Plain text in, XML out
  - Available in Xdebug 2.x

- GDB debugger mode
  - GDB like environment
  - Two output possibilities: text and XML
  - Available in Xdebug 1.3.x and 2.x
  
- DBGp debugger mode
  - Language independent
  - Plain text in, XML out
  - Available in Xdebug 2.x

dbgp:///file:///home/httpd/www.derickrethans.nl/html/include/header.php - ActiveState Komodo - Debugger is in Break Mode

File Edit Code View Debug Project Toolbox Tools Window Help

Projects | Code | Start Page | index.php | header.php

```

$perm_link .= "?item=$key#$key";
} else if (preg_match('/^question_([0-9]*)$/', $key, $m))
$perm_link = 'question.php?id=' . $m[1];
}
?>
<span class='sans'><a href='<?php echo $perm_link; ?>'>permalink</a> | <?
<br />
<?php

```

Enter a new value for \$key:

20041105

Interpret as String  Interpret as Expression

OK Cancel

Breakpoints | Command Output | SCC Output | **Debug: header.php**

Break at /home/httpd/www.derickrethans.nl/html/include/header.php, line 252.

Name	Type	Value
comment_co...	string	no comments
row	bool	0
perm_link	string	month-2004-11.php?item=20041105...
key	string	20041105
month	string	2004-11

Watch | **Locals** | Globals

Call Stack Location

- header.php, line 252, in comments
- index.php, line 63, in {main}

Output | **Call Stack** | HTML

Ready 8859-... Ln: 252 Col: 1 PHP

The screenshot shows the Maguma Workbench 2.1.0-dev interface. The main editor displays a PHP script named 'bug100.php' with the following code:

```

1 <?php
2 function test($i){
3     print $i; // breakpoint's on this line
4 }
5 $j=6;
6 test($j);
7 ?>
    
```

A red arrow indicates a breakpoint is set on line 3. The variable watch window shows the variable '\$i' with a value of 6. The stack trace window shows the current function call: 'function test(\$i)'.

Name	Type	Value
\$i	int	6

The stack trace window shows the following structure:

- Open Files
  - [PHP] bug100.php
    - Global Variables
      - \$j
    - function test(\$i)

The status bar at the bottom indicates the current environment is 'Ins UNIX 3:5'.

The screenshot displays the TruStudio IDE interface with the following components:

- Top Panel:** Contains a toolbar with standard IDE icons (run, stop, refresh, etc.) and a "Variables" window.
- Variables Window:** Lists the following variables:
  - a = "a"
  - l = "l"
  - \$AllowableHTML = <unknown value>
  - \$strip = <unknown value>
  - attrb\_list = "attrb\_list"
- Main Editor:** Shows the source code of `index.php`. A breakpoint is set at line 77, which is highlighted in green. The code includes functions for printing output and checking HTML tags.
 

```

include("footer.php");
}
function print_out ($str) {
    print "test console output";
    return "none";
}
function check_html ($str, $strip="") {
    /* The core of this code has been lifted from phpslash */
    /* which is licenced under the GPL. */
    if ($strip == "nohtml")
        $AllowableHTML=array('');
    $str = print_out($str);
    $str = eregi_replace("<[[:space:]]*([>]*)[[:space:]]*>", '<\1>', $str);
    // Delete all spaces from html tags .
    $str = eregi_replace("<[[:space:]]* img[[:space:]]*([>]*)[[:space:]]*>", '', $str);
    // Delete all img tags
    $str = eregi_replace("<a[>]*href[[:space:]]*=[[:space:]]*\\"?javascript[[:punct:]]*\\"?([>])*>", ''
            
```
- Bottom Panel:** Contains a "Breakpoints" window showing a single breakpoint at `/Users/.../php-proj/TruStudio/html/index.php [line: 77]`.

Share your information

- Protects against infinite recursion
- **Colored** `var_dump()`
- Stacktraces
- Functiontraces
- parameters to functions
- function/methodname, classname
- file:line locations
- memory allocation
- Profiling
- Script execution analysis
- "Remote" debugging



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

DBGp specification:

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

Questions?: [derick@xdebug.org](mailto:derick@xdebug.org)