

# Future of Debugging

Derick Rethans

derick@xdebug.org – @derickr

<https://joind.in/12973>

# Tracing





# Function traces

- Different types: **Textual**, **Parsable** and **HTML**
- Tracing only parts of an application with **xdebug\_start\_trace()** and **xdebug\_stop\_trace()**.
- `xdebug.collect_return`:

```
-> explode('/', '/xdebug-phpnw13/11') /home/httpd/pres2/show2.php:121  
>=> array (0 => '', 1 => 'xdebug-phpnw13', 2 => '11')
```

- `xdebug.collect_assignments`:

```
-> ezcTemplateVariableCollection->__set($name = 'slideNr', $value = '11') /home/httpd/p  
=> $this->variables['slideNr'] = '11' /home/derick/dev/zetacomponents/trunk/Template
```



# Live (Remote) Debugging





# Live Debugging

- DBGp, common Debugging protocol
- Cross-language: PHP, **HHVM**, Python, Perl...
- Komodo (Linux, Windows, Mac – commercial)
- Netbeans (Java-based – free)
- PHPStorm (Java-based – commercial)
- (and many others: <http://xdebug.org/docs/remote#clients> )

# PHP Storm

pres2 - [/home/httpd/pres2] - .../show2.php - PhpStorm 7.1

File Edit View Navigate Code Refactor Run Tools VCS Window Help

pres2 > show2.php

pres2 (/home/htt

- fonts
- images
- locale
- presentations
- reveal.js
- tools
- .gitignore
- .htaccess
- .style-past.css.s
- 1k.js
- 10gen.css

```

Presentation display
$tpl = new ezcTemplate();
$tpl->send->node = $xml->documentElement;
$tpl->send->pres = $this;
$tpl->send->pres->forPdf = isset( $_GET['pdf'] );
$tpl->send->slideNr = $slideNr;
$tpl->send->css = $this->css;

echo $tpl->process( 'slide.ezt' );
}

function renderToRst( $filename )
{
    $tpl = new ezcTemplate();
    $tpl->send->pres = $this;
    echo $tpl->process( 'rst/presentation.ezt' );
}

```

Debug show2.php

Debugger Console

Frames

- show2.php:209, Presentation->display()
- show2.php:142, {main}()

Variables

- \$parts = {array} [3]
- \$slideNr = "13"
- \$this = {Presentation} [6]
  - properties = {array} [12]
  - slideFiles = {array} [15]
    - 0 = "slides/mongodb/title.xml"
    - 1 = "slides/toolbox/me.xml"
    - 2 = "slides/xdebug/agenda-ppp.xml"



# Activating the Remote Debugger

With browser extensions:

The **easiest** Xdebug (Firefox):

<https://addons.mozilla.org/en-US/firefox/addon/the-easiest-xdebug/>

Xdebug Helper (Chrome): <https://chrome.google.com/extensions/detail/eadndfjplgiieldjbigjakmdgkmoaaaoc>

Xdebug Toggler for Safari: <https://github.com/benmatselby/xdebug-toggler>

Xdebug launcher for Opera: <https://addons.opera.com/addons/extensions/details/xdebug-launcher/?display=en>







# Wayback

- Uses traces to store a **full** history of a script run
- Binary, and indexed
- Extensions to DBGp to `step_back` and `seek`
- Playback runner to "act" as PHP/Xdebug
- Browser integration

# Resources



Slides:

<http://derickrethans.nl/talks/debug-future-phpemuc14>

Derick Rethans — @derickr — [derick@xdebug.org](mailto:derick@xdebug.org)

Feedback: <https://joind.in/12973>