

Intro to PHP Debugging

Why? What? How?

The “advanced” session

- Am I in the right place?
- Will there be a test?

Wednesday, January 27, 2010

“Advanced” is a relative term, many shades of grey

Room for Q&A at end, but stop me if you’re lost

Concepts may not “stick” on first view, give it time

Slides and code posted later for review and reference – message boards available too

Common techniques

- `var_dump`
- `print_r`
- `die`

Wednesday, January 27, 2010

Very often mixed and matched

All have the same basic goal of dumping output to the browser while running

```
1 <?php
2
3 ini_set ( 'display_errors', 'On' );
4 error_reporting ( E_ALL );
5
6 $sJsonUrl = 'http://search.twitter.com/trends.json';
7
8 $sJson = file_get_contents ( $sJsonUrl, 0, NULL, NULL );
9 $oTrends = json_decode ( $sJson );
10 $oTrends = process_trends ( $oTrends );
11 $oTrends = filter_trends ( $oTrends );
12 display_trends ( $oTrends, 'chart' );
13
14 echo '<pre>';
15 var_dump ( $oTrends );
16 echo '</pre>';
```

var_dump example

Wednesday, January 27, 2010

nod to carsonified

lines 3 and 4 – general tip (more in resources at end)

bug could be in remote service, in decoding, in processing, in filtering, in display

move var_dump around each time and re-run

```
object(stdClass)#1 (2) {
  ["trends"]=>
  array(10) {
    [0]=>
    object(stdClass)#2 (2) {
      ["name"]=>
      string(11) "#Nowplaying"
      ["url"]=>
      string(48) "http://search.twitter.com/search?q=%23Nowplaying"
    }
    [1]=>
    object(stdClass)#3 (2) {
      ["name"]=>
      string(5) "Haiti"
      ["url"]=>
      string(52) "http://search.twitter.com/search?q=Haiti+OR+%23haiti"
    }
    [2]=>
    object(stdClass)#4 (2) {
      ["name"]=>
      string(11) "#whatislove"
      ["url"]=>
      string(48) "http://search.twitter.com/search?q=%23whatislove"
    }
  }
}
```

var_dump output

Wednesday, January 27, 2010

Clipped for brevity
shows basic JSON structure
recursive

```
stdClass Object
(
  [trends] => Array
  (
    [0] => stdClass Object
    (
      [name] => #Nowplaying
      [url] => http://search.twitter.com/search?q=%23Nowplaying
    )
    [1] => stdClass Object
    (
      [name] => Haiti
      [url] => http://search.twitter.com/search?q=Haiti+OR+%23haiti
    )
    [2] => stdClass Object
    (
      [name] => #whatislove
      [url] => http://search.twitter.com/search?q=%23whatislove
    )
    [3] => stdClass Object
```

print_r output

Wednesday, January 27, 2010

slightly less information
little more whitespace
can also be captured by a variable

Problems

- Time consuming
- Over-simplified
- Alteration of code

Wednesday, January 27, 2010

Trial and error . . . move `var_dump` around, comment/un-comment

A single snapshot in time, no interactivity, history (what if you want only 3rd loop – more code)

You have to add code (`var_dump`) to debug code, can pile up in a longer debug session

Why do it different?

- Improved time to resolution
- Interactivity
- Clean, repeatable

Wednesday, January 27, 2010

The “why” question – answers all the shortcomings discussed

Far less trial and error – step through code line by line

Watch values change over time or force a change as desired

Nothing littered in code, IDEs allow you to easily unset all breaks

Use the right tool for the job, var_dump has its place (remote troubleshooting, no IDE)

What is the alternative?

- IDE based debugger
- Breakpoints
- Stepping in/over/out of code
- Watches

Wednesday, January 27, 2010

the “what” question

Zend Studio, Eclipse+XDebug

go over basic concept of interactive debugger: interactivity, real time response, IDE integration

define common terms

How does it work?

- DEMO time

Resources

- <http://carsonified.com/blog/dev/how-to-debug-in-php/>
- <http://devzone.zend.com/article/2803>
- <http://aaronsaray.com/blog/2008/05/20/xdebug-and-eclipse-pdt-on-windows-from-start-to-finish/>
- <http://krumo.sourceforge.net/>

Wednesday, January 27, 2010

Good general tips – even before breaking out the debugger (error level, logs, defaults)
5-part series, covers many related topics – xdebug specific
Helpful tutorial for setting up the environment
Nice enhancements to var_dump-like functionality (xdebug pretties)

Conclusion

- Give it a try
- Choose your battles
- Ask for help

Wednesday, January 27, 2010

The best way to see the power is to try it yourself, especially when facing a tough bug
Use the right tool for the job
Discussion board, email, future sessions