



# PHP performance hints.

Александр Веремьев (Zend Framework core team)  
[alexander.v@zend.com](mailto:alexander.v@zend.com)

# Передача строк по ссылке

---

# Передача строк по ссылке

---

- `function test1($str) {`
- `$chCounter = 0;`
- 
- `for ($count = 0; $count < strlen($str); $count++) {`
- `if ($str[$count] == '4') $chCounter++;`
- `}`
- 
- `return $chCounter;`
- `}`

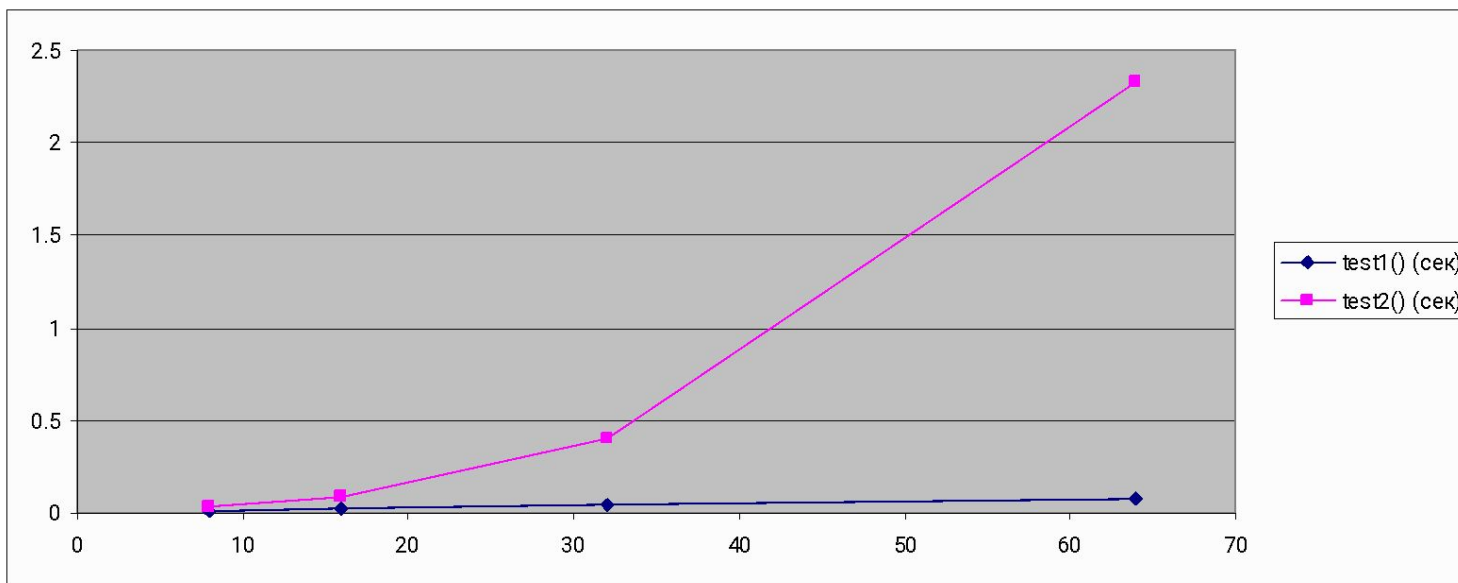
# Передача строк по ссылке

---

- `function test2(&$str) {`
- `$chCounter = 0;`
- 
- `for ($count = 0; $count < strlen($str); $count++) {`
- `if ($str[$count] == '4') $chCounter++;`
- `}`
- 
- `return $chCounter;`
- `}`

# Передача строк по ссылке

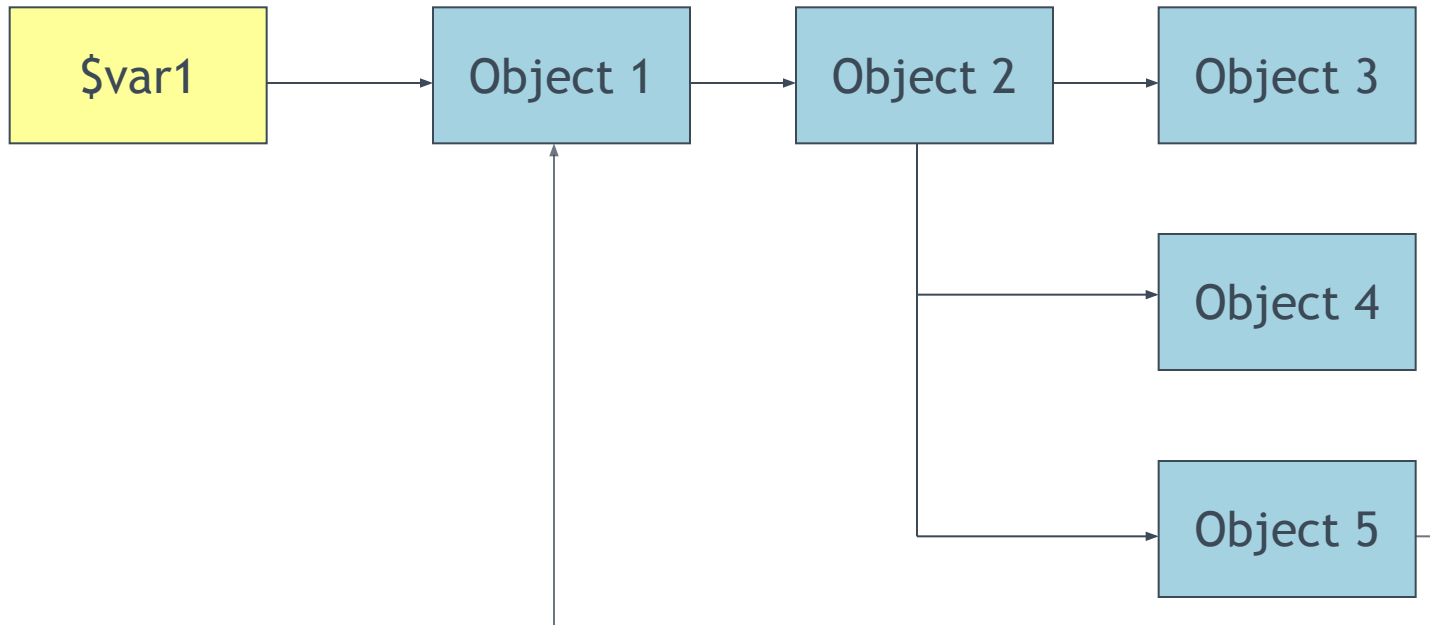
Размер строки (Мб)	test1() (сек)	test2() (сек)
8	0.01	0.03
16	0.02	0.09
32	0.04	0.4
64	0.08	2.33



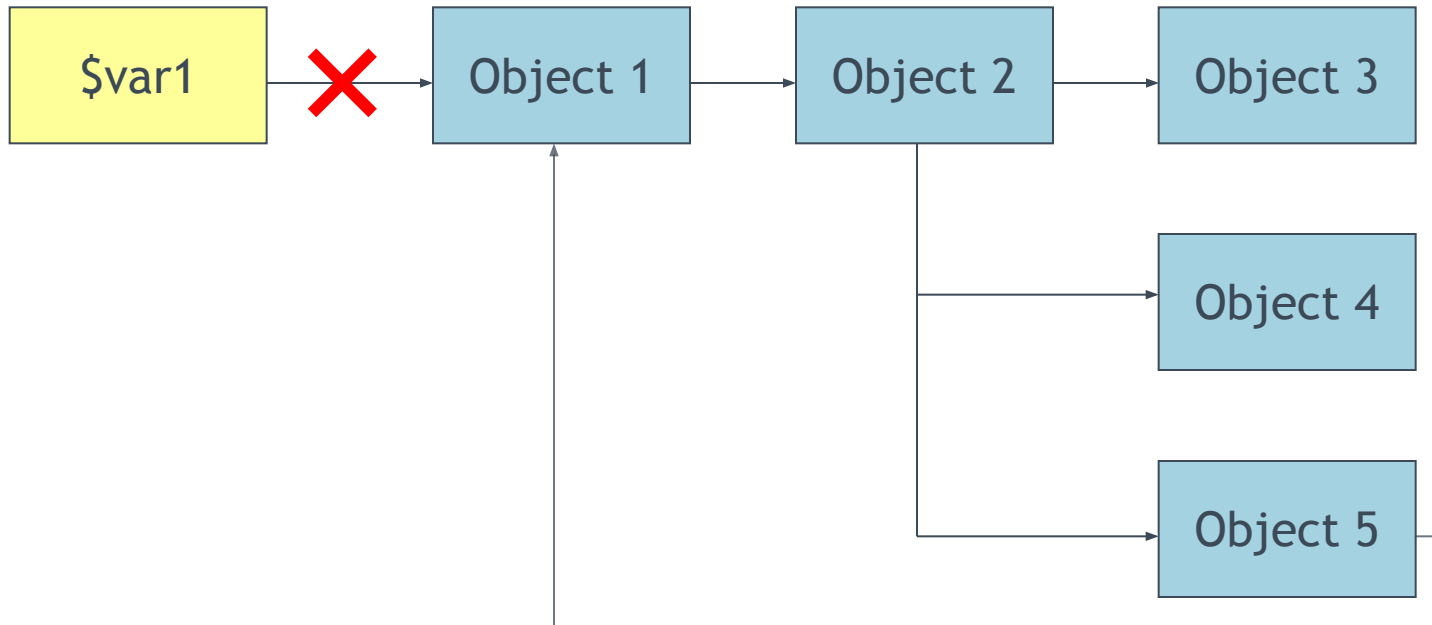
# Циклические ссылки

---

# Циклические ссылки в структурах объектов



# Циклические ссылки в структурах объектов

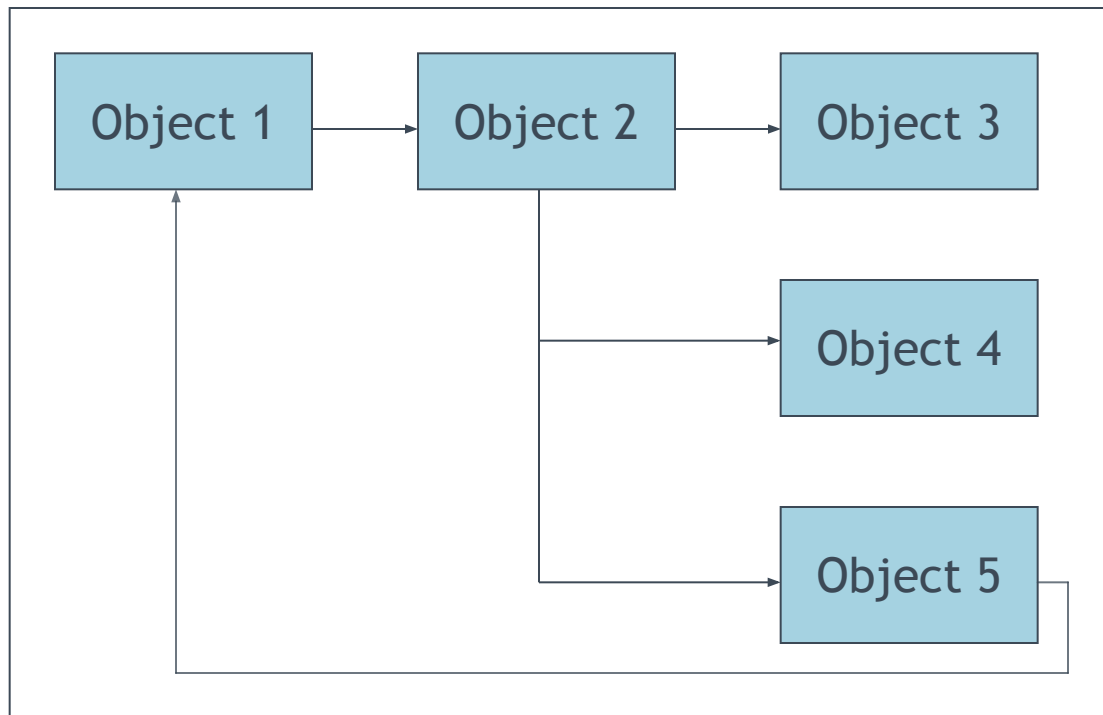


```
unset($var1);
```



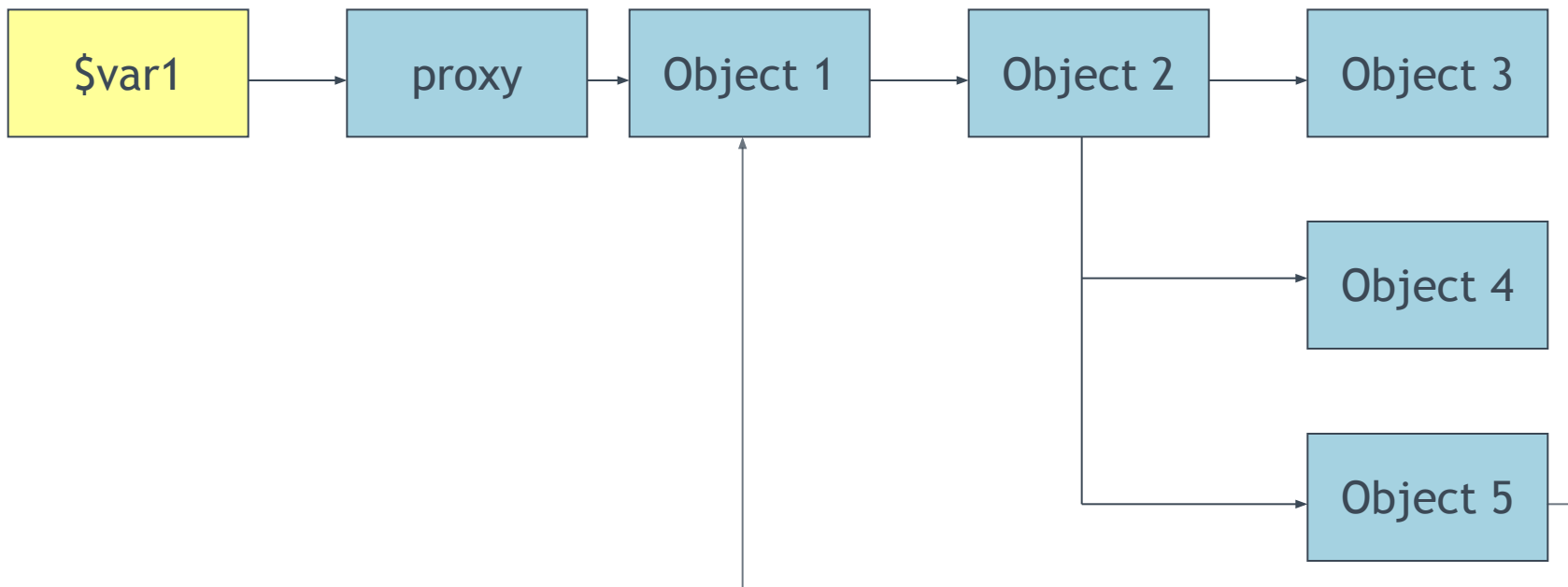
# Циклические ссылки в структурах объектов

\$var1



```
unset($var1);
```

# Циклические ссылки в структурах объектов



.= оператор

# . = оператор и обработка больших строк

---

- `$str = "";`
- 
- `while ($condition) {`
- `...`
- `$str .= $something;`
- `}`



# . = оператор и обработка больших строк

---

- \$str = "";
- 
- while (\$condition) {
- ...
- \$str .= \$something;
- }



# . = оператор и обработка больших строк

---

- `$str = "";`
- 
- `while ($condition) {`
- `...`
- `$str .= $something;`
- `}`



# .= оператор и обработка больших строк

---

- \$str = "";
- 
- while (\$condition) {
- ...
- \$str .= \$something;
- }



# . = оператор и обработка больших строк

---

- `$str = "";`
- 
- `while ($condition) {`
- `...`
- `$str .= $something;`
- `}`





# . = оператор и обработка больших строк

---

- `$substrings = array();`
- `while ($condition) {`
- `...`
- `$substrings[] = $something;`
- `}`
- `$str = implode($substrings);`

# Неэффективность `serialize()/unserialize()`

---

# Неэффективность `serialize()/unserialize()`

---

```
array(134, 'test stri\ng', array('float test', 0.0348), 456)
```

=>

```
a:4:{i:0;i:134;i:1;s:12:"test stri'ng";i:2;a:2:{i:0;s:10:"float test";i:1;d:0.034799999999999997601918266809661872684955596923828125;}i:3;i:456;}
```

# Неэффективность `serialize()/unserialize()`

```
array(134, 'test stri\ng', array('float test', 0.0348), 456)
```

=>

```
a:4:{i:0;i:134;i:1;s:12:"test stri'ng";i:2;a:2:{i:0;s:10:"float test";i:1;d:0.0347999999999999997601918266809661872684955596923828125;}i:3;i:456;}
```

```
array(134, 'test stri\ng', array('long test', 34823734), 456)
```

=>

```
a:4:{i:0;i:134;i:1;s:12:"test stri'ng";i:2;a:2:{i:0;s:9:"long test";i:1;i:34823734;}i:3;i:456;}
```

# Ручной “method inlining”

---

# Замена объектов массивами

---

# Вопросы

---

Александр Веремьев  
alexander.v@zend.com