HipHop Compiler for PHP

Transforming PHP into C++

HipHop Compiler Team
Facebook, Inc.
May 2010
<?php

function tally($count) {
    $sum = 0;
    for ($i = 0; $i < $count; ++$i) {
        $sum += $i;
    }
    return $sum;
}

print tally(10) . "\n";
<?php

class Tool extends Object {
    public $name;

    public use($target) {}
}

$tool = new Tool();
$tool->name = 'hammer';
$tool->use($nail);
PHP Statements and Expressions


ExpressionList, AssignmentExpression, SimpleVariable, DynamicVariable, StaticMemberExpression, ArrayElementExpression, DynamicFunctionCall, SimpleFunctionCall, ScalarExpression, ObjectPropertyExpression, ObjectMethodExpression, ListAssignment, NewObjectExpression, UnaryOpExpression, IncludeExpression, BinaryOpExpression, QOpExpression, ArrayPairExpression, ClassConstantExpression, ParameterExpression, ModifierExpression, ConstantExpression, EncapsListExpression,
<?php
$a = 12345;
$a = "hello";
$a = array(12345, "hello", array());
$a = new Object();

$c = $a + $b; // integer or array
$c = $a . $b; // implicit casting to strings
Core PHP library is small

- Most are in functional style
- ~200 to 500 basic functions

```php
<?php

$len = strlen("hello"); // C library
$ret = curl_exec($curl); // open source
```
<?php

function tally($count) {
    $sum = 0;
    for ($i = 0; $i < $count; ++$i) {
        $sum += $i;
        var_dump($sum);
    }
    return $sum;
}

PHP is easy to learn

- easy to read
- easy to write
- easy to debug

Hello, World!
PHP is slow 😞

http://shootout.alioth.debian.org/u64q/benchmark.php?test=all&lang=all
Why is Zend Engine slow?

- Byte-code interpreter
- Dynamic symbol lookups
  - functions, variables, constants
  - class methods, properties, constants
- Weakly typing
  - zval
  - array()
Transforming PHP into C++

- g++ is a native code compiler
- static binding
  - functions, variables, constants
  - class methods, properties, constants
- type inference
  - integers, strings, arrays, objects, variants
  - struct, vector, map, array