Software Testing on the Web

Martin Schütte

Outline

Main Questions

How can I know (my) software is correct?
How does my boss know software is correct?
How can we discuss what “correct” is, anyway?

Main Questions

How can I know (my) software is correct?
How does my boss know software is correct?
How can we discuss what “correct” is, anyway?

We always need:

- implicit assumptions,
- explicit specifications.

Walking on water and developing software from a specification are easy if both are frozen.

– Edward V. Berard

> *Walking on water and developing software from a specification are easy if both are frozen.*
>
> – Edward V. Berard

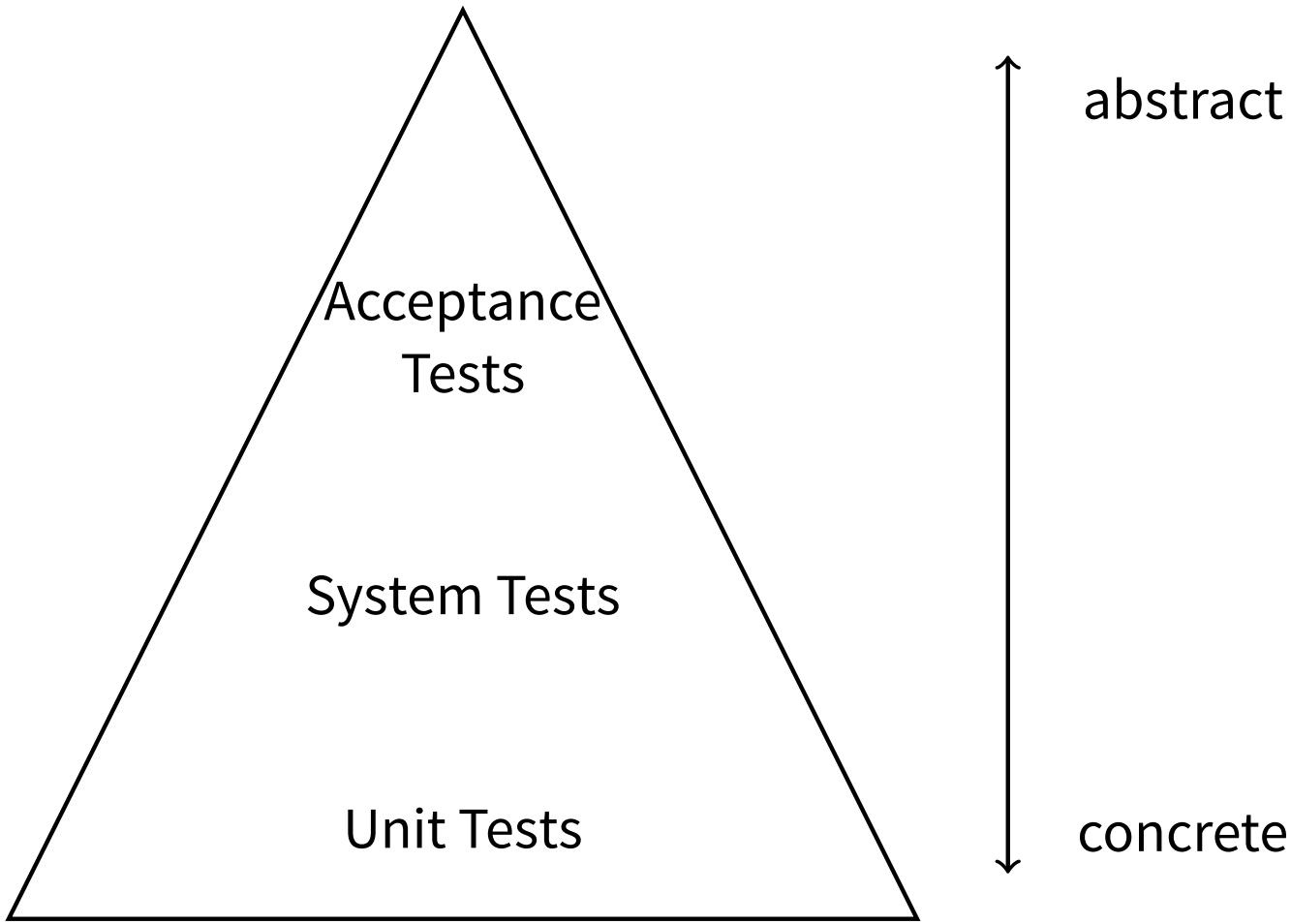
Let's just update PHP…

Let's just change the DB schema…

Let's just add feature *X*…

Is the software still correct?

Levels of Testing



Regression Testing

Bug Tracker

Description:
Put code samples in the "Test script" section **below** and upload patches **below**.

found a bug in DTA::getNum()

Test script:
A short test script you wrote that demonstrates the bug. Please **do not** post more than 20 lines of code. If the code is longer than 20 lines, provide a URL to the source code or attach a patch that will reproduce the bug.

```
$d = new DTA();
$input = "12345";
print $d->getNum($input, 3, 4);
```

Expected result:
What do you expect to happen or see when you run the test script above?

45

Actual result:
This could be a backtrace for example. Try to keep it as short as possible without leaving anything relevant out.

45.001

Send bug report

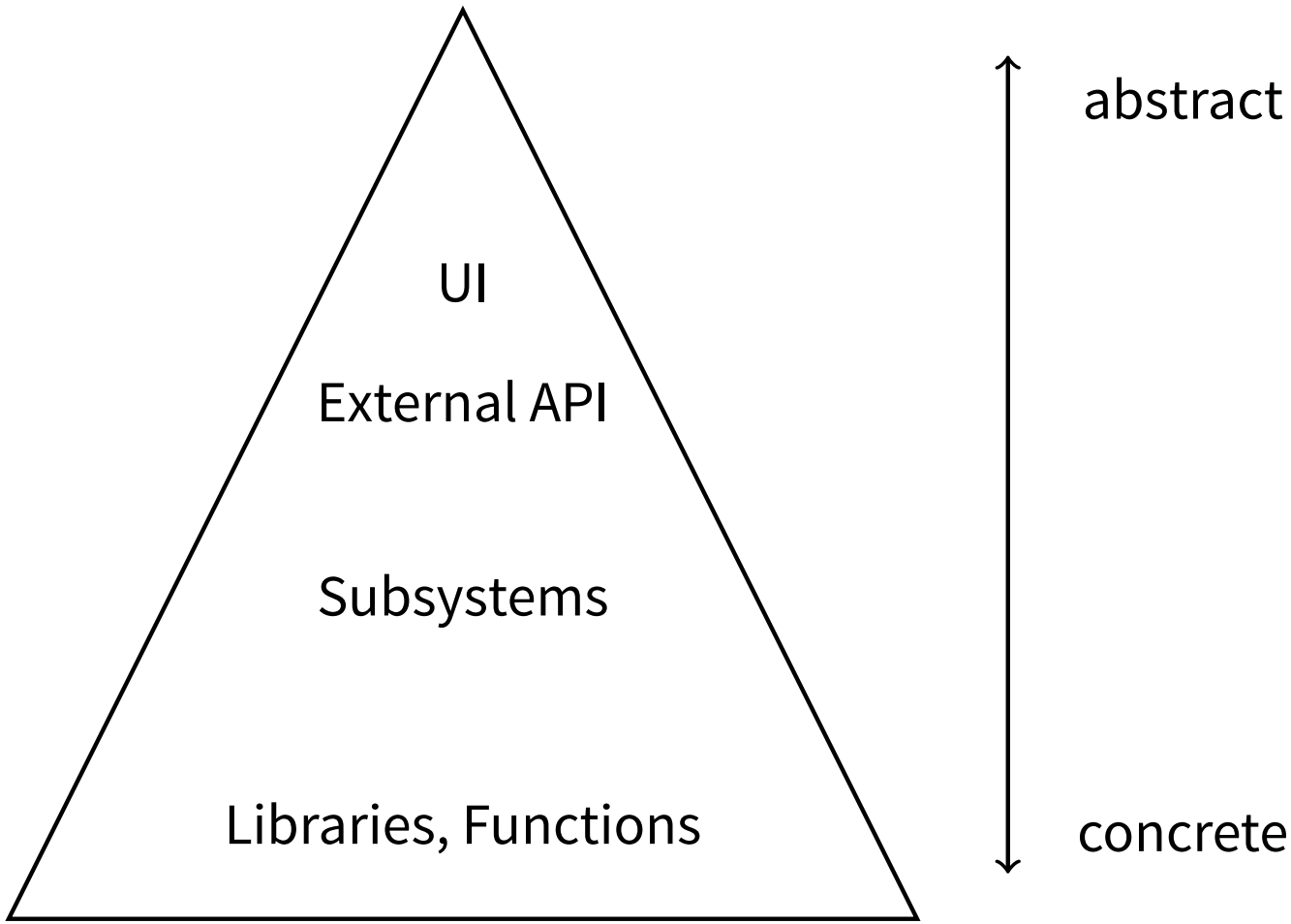
PHPUnit

```php
class DTABaseTest extends PHPUnit_Framework_TestCase
{
    protected function setUp()
    {
        $this->fixture = new DTABase();
    }

    public function testGetNumTooShort()
    {
        $input = "12345";
        $off = 3;
        $len = 4;
        $rc = $this->fixture->getNum($input, $off, $len);
        $this->assertEquals("45", $rc);
    }
    // ...
}
```

PHPUnit

```
~/Payment_DTA> phpunit tests/DTABaseTest.php
PHPUnit 3.6.11 by Sebastian Bergmann.

F....

Time: 0 seconds, Memory: 3.75Mb

There was 1 failure:

1) DTABaseTest::testGetNumTooShort
Failed asserting that 45.001 matches expected '45'.

/usr/home/mschuett/Payment_DTA/tests/DTABaseTest.php:50
/usr/local/bin/phpunit:46

FAILURES!
Tests: 5, Assertions: 5, Failures: 1.
```

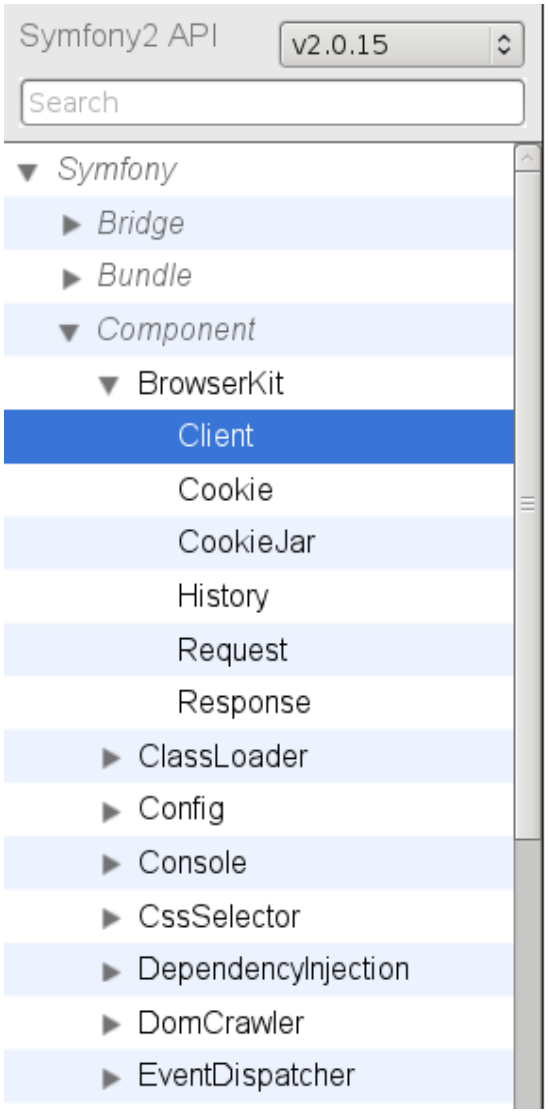
Levels of Documentation



PHPdoc/JavaDoc/Doxygen etc.

```php
/**
 * Gets single server parameter for specified key.
 *
 * @param string $key A key of the parameter to get
 * @param string $default A default value when key is undefined
 *
 * @return string A value of the parameter
 */
public function getServerParameter($key, $default = '')
{
    return (isset($this->server[$key]))
        ? $this->server[$key] : $default;
}
```

Low Level Documentation



public string getServerParameter(string $key, string $default = '') *at line 130*

Gets single server parameter for specified key.

Parameters

string	$key	A key of the parameter to get
string	$default	A default value when key is undefined

Return Value

string	A value of the parameter

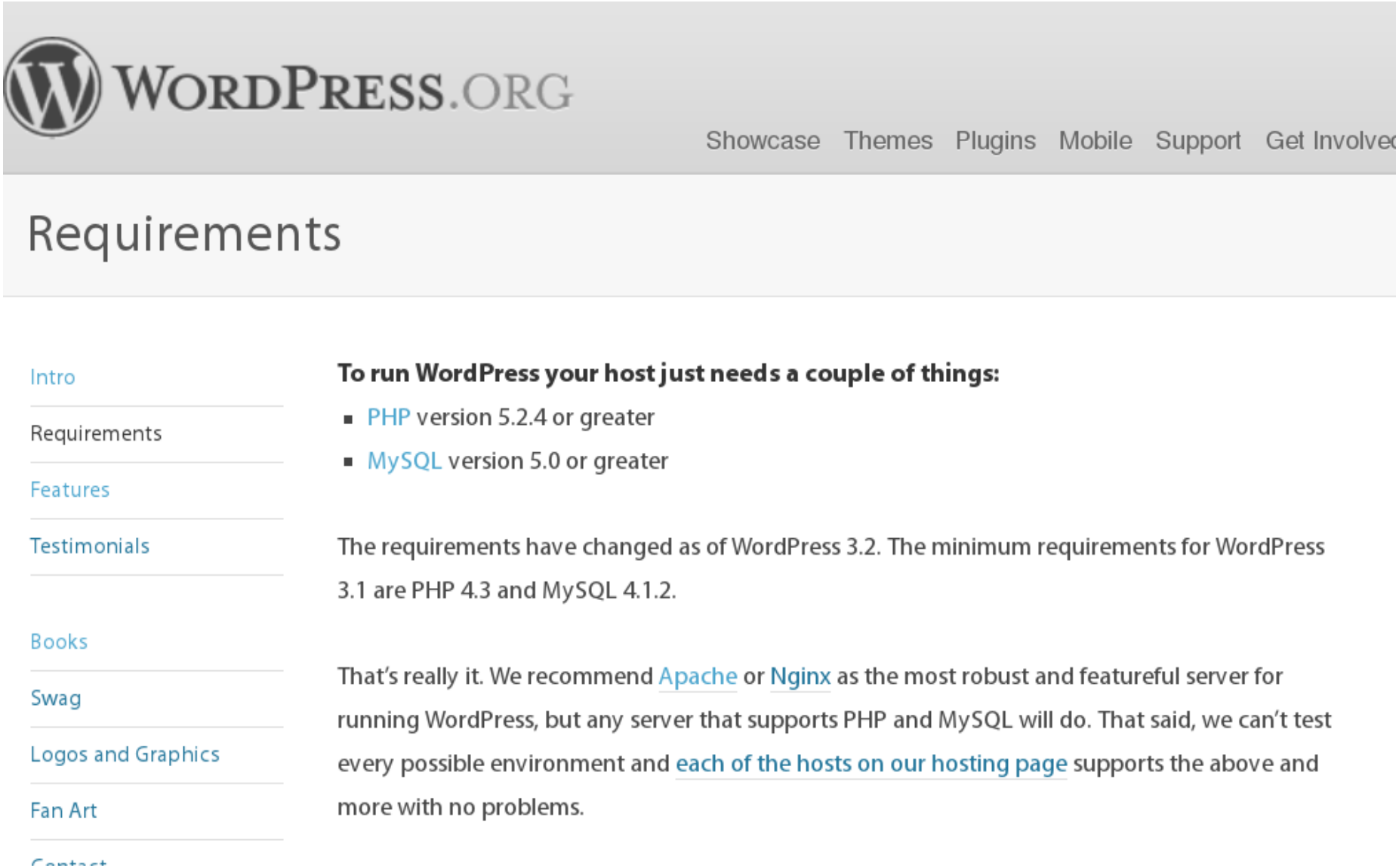
public History getHistory() *at line 142*

Returns the History instance.

Return Value

History	A History instance

High Level Documentation

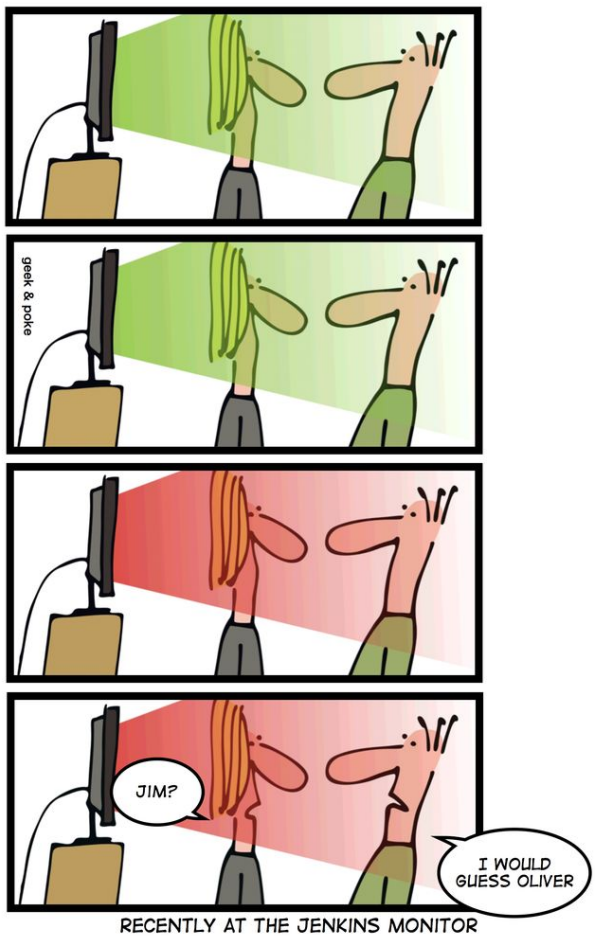


Tests are Documentation

```
~/Payment_DTA> phpunit --testdox tests/DTABaseTest.php
PHPUnit 3.6.11 by Sebastian Bergmann.

DTABase
 [ ] Get num too short
 [x] Get num offset too big
 [x] Get str too short
 [x] Get str offset too big
 [x] Get str invalid
```

Things to build upon Automated Testing

- Code coverage and software metrics
- `git bisect run phpunit`:
  find commit that introduces a bug
- Continuous Integration:
  - frequent check-ins to mainline branch
  - verified by automated builds and tests
- Continuous Deployment:
  - automated deployment to production
  - useful: automated rollback



Example: PHPUnit Code Coverage

class-http.php

Current file: /vagrant/trunk/wordpress/wp-includes/class-http.php

Legend: executed | not executed | dead code

	Coverage								
	Classes		Functions / Methods			CRAP		Lines	
Total	0.00%	0 / 7	25.64%	10 / 39		CRAP		48.71%	339 / 696
WP_Http	0.00%	0 / 1	33.33%	4 / 12		412.20		63.38%	135 / 213
request($url, $args = array()			0.00%	0 / 1		37.01		76.06%	54 / 71
_get_first_available_transport($args, $url = null)			0.00%	0 / 1		3.14		75.00%	6 / 8
_dispatch_request($url, $args)			100.00%	1 / 1		4		100.00%	11 / 11
post($url, $args = array()			0.00%	0 / 1		2		0.00%	0 / 3
get($url, $args = array()			0.00%	0 / 1		2		0.00%	0 / 3
head($url, $args = array()			100.00%	1 / 1		1		100.00%	3 / 3
processResponse($strResponse)			100.00%	1 / 1		2		100.00%	2 / 2
processHeaders($headers)			0.00%	0 / 1		11.20		88.24%	30 / 34
buildCookieHeader(&$r)			0.00%	0 / 1		7.23		22.22%	2 / 9
chunkTransferDecode($body)			0.00%	0 / 1		42		0.00%	0 / 20
block_request($uri)			0.00%	0 / 1		104.20		8.33%	2 / 24

Example: Jenkins CI

Jenkins

search | log in

Jenkins › SpamAssassin-trunk | ENABLE AUTO REFRESH

- Back to Dashboard
- Status
- Changes
- Workspace
- Subversion Polling Log

Build History (trend)

#8084	08-Oct-2012 16:13:00	89MB
#8083	08-Oct-2012 08:11:07	88MB
#8082	08-Oct-2012 06:13:01	89MB
#8081	07-Oct-2012 18:15:48	90MB
#8080	07-Oct-2012 08:13:13	89MB
#8079	07-Oct-2012 06:13:15	89MB
#8078	06-Oct-2012 10:13:08	90MB
#8077	06-Oct-2012 08:11:05	89MB
#8076	06-Oct-2012 04:22:42	89MB
#8075	05-Oct-2012 10:14:44	90MB
#8074	05-Oct-2012 08:43:29	90MB
#8073	05-Oct-2012 04:51:18	89MB
#8072	04-Oct-2012 20:15:47	89MB
#8071	04-Oct-2012 14:18:32	90MB
#8070	04-Oct-2012 10:12:02	89MB
#8069	04-Oct-2012 08:12:21	90MB

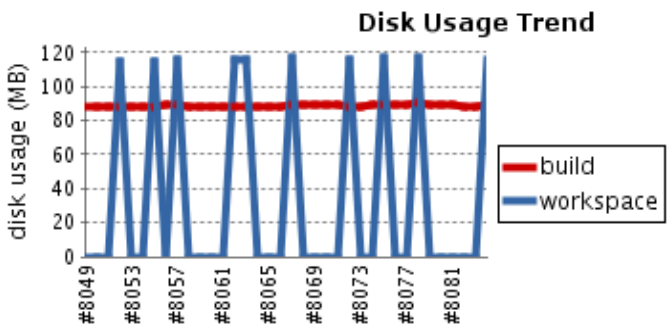
Project SpamAssassin-trunk

SpamAssassin

Builds and runs the SpamAssassin test suite, using SVN trunk.

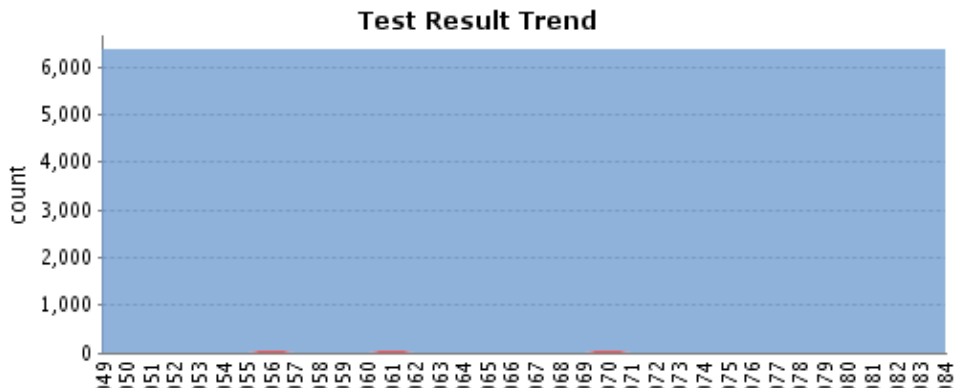
- Workspace
- Last Successful Artifacts
- Recent Changes
- Latest Test Result (no failures)

Disk Usage: Workspace 116MB, Builds 3GB



Permalinks

- Last build (#8084), 5 hr 5 min ago
- Last stable build (#8084), 5 hr 5 min ago
- Last successful build (#8084), 5 hr 5 min ago
- Last unstable build (#8070), 4 days 11 hr ago
- Last unsuccessful build (#8070), 4 days 11 hr ago



Example: Travis CI for GitHub

pear/Payment_DTA

Current | Build History | Pull Requests | Branch Summary

Build	2	Commit	5402044 (trunk)
Finished	37 minutes ago	Compare	d8667c610e5c...5402044e548c
Duration	53 sec	Author	Martin Schuette
		Committer	Martin Schuette
Message	use travis-ci.org		

Build Matrix

Job	Duration	Finished	Php
2.1	19 sec	37 minutes ago	5.2
2.2	16 sec	39 minutes ago	5.3
2.3	18 sec	37 minutes ago	5.4

Approaches to Unit Testing

- Exploratory testing (e. g. for 3rd party libs)
- Regression testing (for found & fixed bugs)
- Spike and Stabilise testing (after code, before refactoring)
- Test Driven Development (TDD): test first, then code

Program testing can be used to show the presence of bugs, but never to show their absence!

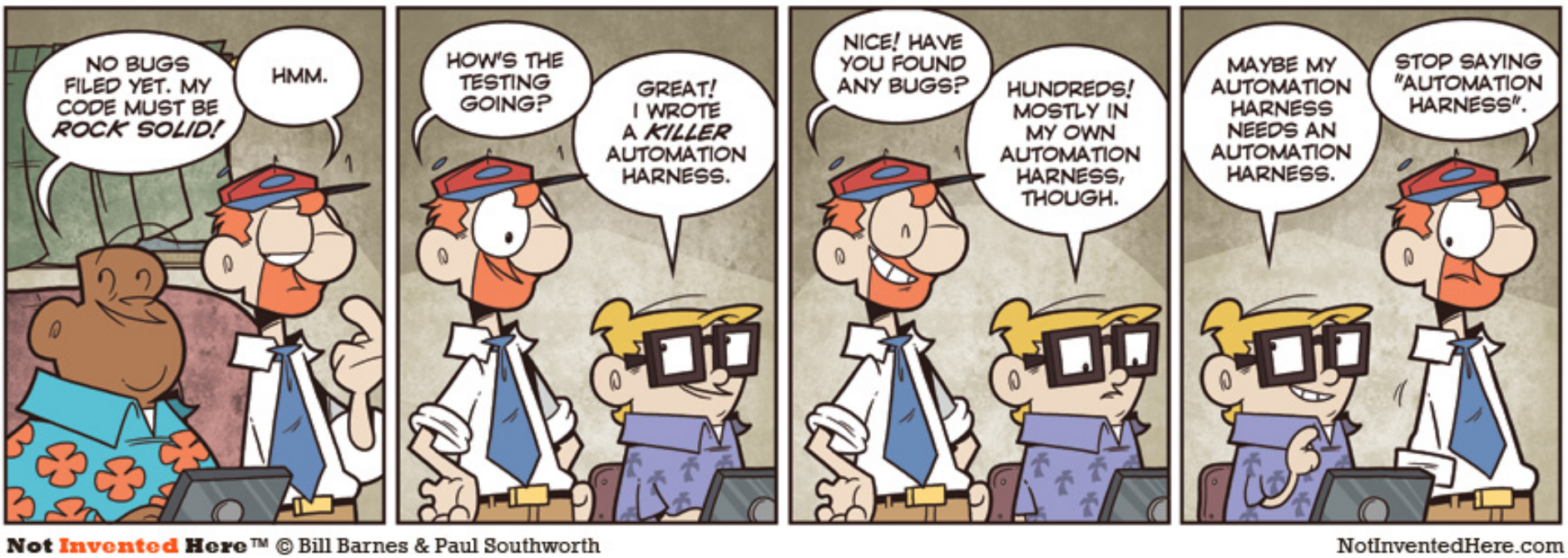
– Edsger Dijkstra,
Notes on Structured Programming

Testing by itself does not improve software quality.
Test results are an indicator of quality, but in and of themselves, they don't improve it. […]
If you want to improve your software, don't test more; develop better.

– Steve McConnell, Code Complete

Cost of Testing

- Testing (like Documentation) has a cost
- often: productivity improvement > cost
- but ROI depends on type of project/software



HTTP API

```php
class TumblrAPITest extends PHPUnit_Framework_TestCase
{
    public function testPostDatetime()
    {
        $url = "http://staff.tumblr.com/api/read";
        $ch  = curl_init($url);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
        $xmlsrc = curl_exec($ch);

        $xmlobj = new SimpleXMLElement($xmlsrc);
        $str_date = date_timestamp_get(date_create(
            $xmlobj->posts->post[0]['date-gmt']));
        $unix_date = intval(
            $xmlobj->posts->post[0]['unix-timestamp']);

        $this->assertEquals($str_date, $unix_date);
    }
}
```

HTML-Content

```php
class BlitTest extends PHPUnit_Framework_TestCase
{
    public function testSiteTitle()
    {
        $url = "http://blit.org/";
        $ch  = curl_init($url);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
        curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);
        $xmlsrc = curl_exec($ch);

        $dom    = @DOMDocument::loadHTML($xmlsrc);
        $title  = $dom->getElementsByTagName('title')
                      ->item(0)->textContent;

        $this->assertContains('BLIT 2012', $title);
    }
}
```

HTML-Content

```php
class BlitTest extends PHPUnit_Framework_TestCase
{
    public function testSiteTitle()
    {
        $url = "http://blit.org/";
        $ch  = curl_init($url);
        curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
        curl_setopt($ch, CURLOPT_FOLLOWLOCATION, 1);
        $xmlsrc = curl_exec($ch);

        $dom    = @DOMDocument::loadHTML($xmlsrc);
        $title  = $dom->getElementsByTagName('title')
                      ->item(0)->textContent;

        $this->assertContains('BLIT 2012', $title);
    }
}
```

Goutte

a simple PHP Web Scraper

```php
class BlitTest extends PHPUnit_Framework_TestCase
{
    public function testSiteTitle()
    {
        $client  = new Client();
        $crawler = $client
            ->request('GET', 'http://blit.org/');
        $title   = $crawler
            ->filter('html head title')
            ->first()->text();

        $this->assertContains('BLIT 2012', $title);
    }
}
```

Selenium

a suite of tools to automate web browsers across platforms

```
class WpTest extends PHPUnit_Extensions_SeleniumTestCase
{
    public function setUp()
    {
        $this->setHost("localhost");
        $this->setBrowser("firefox");
        $this->setBrowserUrl("http://blit.org/");
    }
    public function testSiteTitle()
    {
        $this->open("/");
        $this->assertText("//html/head/title",
            "*BLIT 2012*");
    }
```

Selenium, contd.

```
    public function testSearch()
    {
        $this->open(”/blog/”);
        $this->type(”id=s”, ”keysigning”);
        $this->clickAndWait(”id=searchsubmit”);

        $this->verifyText(”css=#content”,
            ”regexp:.*Keysigning auf dem 9. BLIT.*”);
    }
}
```

Selenium, PHPUnit command line

```
[mschuett@dagny] ~/blit/code% phpunit selenium_test.php
PHPUnit 3.7.6 by Sebastian Bergmann.

..

Time: 19 seconds, Memory: 3.25Mb

OK (2 tests, 2 assertions)
```

Selenium, screenshot feature

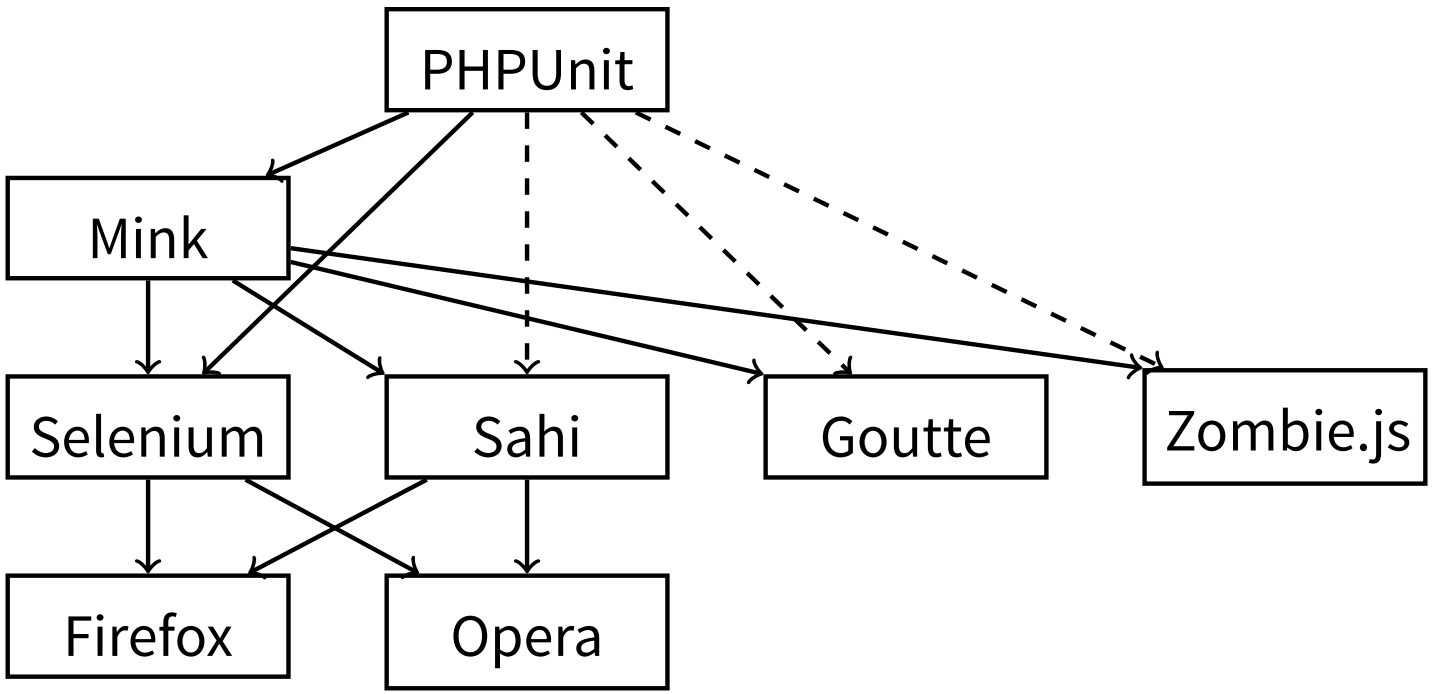
```
class BlitTest extends PHPUnit_Extensions_SeleniumTestCase {
    protected $captureScreenshotOnFailure = TRUE;
    protected $screenshotPath = '/tmp/screenshots';
    protected $screenshotUrl = 'http://localhost/screenshots';

    public function setUp()
    { /* ... */ }

    public function testSearch()
    {
        $this->open("/blog/");
        $this->type("id=s", "social");
        $this->clickAndWait("id=searchsubmit");
        $this->verifyText("css=#content",
            "regexp:.*Social Event.*");
    }
}
```

Selenium, screenshot feature

```
[mschuett@dagny] ~/blit/code% phpunit selenium_test_2.php
PHPUnit 3.7.6 by Sebastian Bergmann.

E

Time: 11 seconds, Memory: 3.50Mb

There was 1 error:

1) BlitTest::testSearch
Current URL: http://blit.org/blog/wordpress/?s=social
Screenshot: http://localhost/screenshots/d88a3c7f1ca4350213805a22744e0b5c.

Failed command: verifyText('css=#content', 'regexp:.*Social Event.*')
Failed asserting that 'Ihre Suche ergab keine Treffer. ...'
    matches PCRE pattern "/.*Social Event.*/".

FAILURES!
Tests: 1, Assertions: 1, Errors: 1.
```

d88a3c7f1ca4350213805a22744e0b5c.png

The Big Picture



Problems with Web Drivers

Differences in:

- web functionality (HTML, CSS, JS, AJAX)
  $\Rightarrow$ performance
- selectors (ID, CSS, XPath, DOM)
- test functionality and integration with test framework

Recommendation: Select two Drivers,

1. simple and fast one for unit testing and
2. Selenium/Sahi for integration and UX testing.

Testing in Production = Monitoring

Example: check webshop availability

Possible layers:

- ping server
- check HTTP status ”200 OK”
- check homepage title
- login as user
- complete order process

The next level: Behaviour Tests

```
Funktionalität: Selbstpräsentation
    Um etwas über den BLIT zu erfahren
    Als Besucher
    Möchte ich Infos im Blog finden

    Szenario: Keysigning suchen
        Angenommen ich bin auf "/blog"
        Und ich suche nach "keysigning"
        Dann ich sollte "Keysigning auf dem 9. BLIT" sehen
```

Next Level: Behaviour Driven Development

TDD with other syntax?

Is BDD the same as TDD? Yes. If you're a programmer, and your entire team is programmers, and all your stakeholders are programmers.

– Dan North

We don't call them "acceptance tests" because you can't ask a business person "Please help me with my acceptance test". Try "I'd like to talk to you about the scenario where…" instead. Or, "Can you give me an example?" Either of these are good.

– Liz Keogh

Unit & Behaviour Testing

Unit Tests

- unit testing
- programmers
- programming language
- bottom-up
- `assertXYZ`
- tests derived from user stories

$\Rightarrow$ development tool

Behaviour Tests

- ~~acceptance test~~ scenarios
- non-developers
- language of business domain
- top-down / outside-in
- *X* should do *Y*
- execute user stories

$\Rightarrow$ communication tool

One Notation/“Language”: Gherkin

```
Feature: [one line describing the story]
  In order to [benefit]
  As a [role]
  I want [feature]

   Scenario: [A scenario specific goal]
    Given [Something that needs to have happened]
    When [Some task I must do]
    And  [Another task]
    Then [Some way I know I've achieved my goal]
```

I18N

```
# language: de

Funktionalität: [one line describing the story]
  Um zu erreichen [benefit]
  Als [role]
  Möchte ich [feature]

  Szenario: [Title]
    Angenommen [context]
    Wenn [event]
    Dann [outcome]
    Und  [another outcome]
```

I18N

```
# language: ru

Функционал: [description]

  Сценарий: [Title]
    Допустим [context]
    И [more context]
    Если [event]
    Тогда [outcome]
```

```
# language: en-au

Crikey: [description]

  Mate: [Title]
    Ya know how [context]
    N [more context]
    When [event]
    Ya gotta [outcome]
```

TDD with other syntax?

```
class TestList(object):
    def test_empty_list_is_false(self):
        list = []
        assertEqual(bool(list), False)

    def test_populated_list_is_true(self):
        list = []
        list.append('item')
        assertEqual(bool(list), True)
```

TDD with other syntax?

```
Scenario: New lists are empty
  Given a new list
  Then the list should be empty.

Scenario: Lists with things in them are not empty.
  Given a new list
  When we add an object
  Then the list should not be empty.
```

TDD with other syntax?

```
Scenario: New lists are empty
  Given a new list
  Then the list should be empty.

Scenario: Lists with things in them are not empty.
  Given a new list
  When we add an object
  Then the list should not be empty.
```

Implementation / Translation

A look behind the curtain:

- framework is clever but not magical
- some translation needed
- statements have to become executable code

Mechanism:

- plain sentence $\rightarrow$ method name
- quoted words $\rightarrow$ arguments
- matching with annotated RegExp
- methods yield success, exception, or pending exception

Behat/Mink FeatureContext.php

```php
class FeatureContext extends BehatContext
{
    /**
     * Initialize.
     */
    public function __construct(array $parameters)
    {
    }

    /**
     * @Given /^I have done something with "([^"]*)"$/
     */
    public function iHaveDoneSomethingWith($argument)
    {
        doSomethingWith($argument);
    }
}
```

Behat/Mink FeatureContext.php

```php
use Behat\MinkExtension\Context\MinkContext;

class FeatureContext extends BehatContext
{
    public function __construct(array $parameters)
    {
        $this->useContext('mink', new MinkContext());
    }
}
```

`MinkContext` adds web-related methods and their translations, e. g.:

- `visit($page)`
- `fillField($field, $value)`
- `pressButton($button)`

Behat Output

```
[mschuett@dagny] ~/blit/code% ./vendor/bin/behat
Funktionalität: Selbstpräsentation
  Um etwas über den BLIT zu erfahren
  Als Besucher
  Möchte ich Infos im Blog finden

  Szenario: Keysigning suchen                                  # features/blit.fea
    Angenommen ich bin auf ”/blog”                             # Behat\MinkExtensi
    Und ich suche nach ”keysigning”
    Dann ich sollte ”Keysigning auf dem 9. BLIT” sehen # Behat\MinkExtensi

1 scenario (1 undefined)
3 steps (1 passed, 1 skipped, 1 undefined)
0m0.868s

...
```

Behat Output, contd.

```
...

You can implement step definitions for undefined steps
    with these snippets:

    /**
     * @Given /^ich suche nach "([^"]*)"$/
     */
    public function ichSucheNach($arg1)
    {
        throw new PendingException();
    }
```

Edit FeatureContext.php

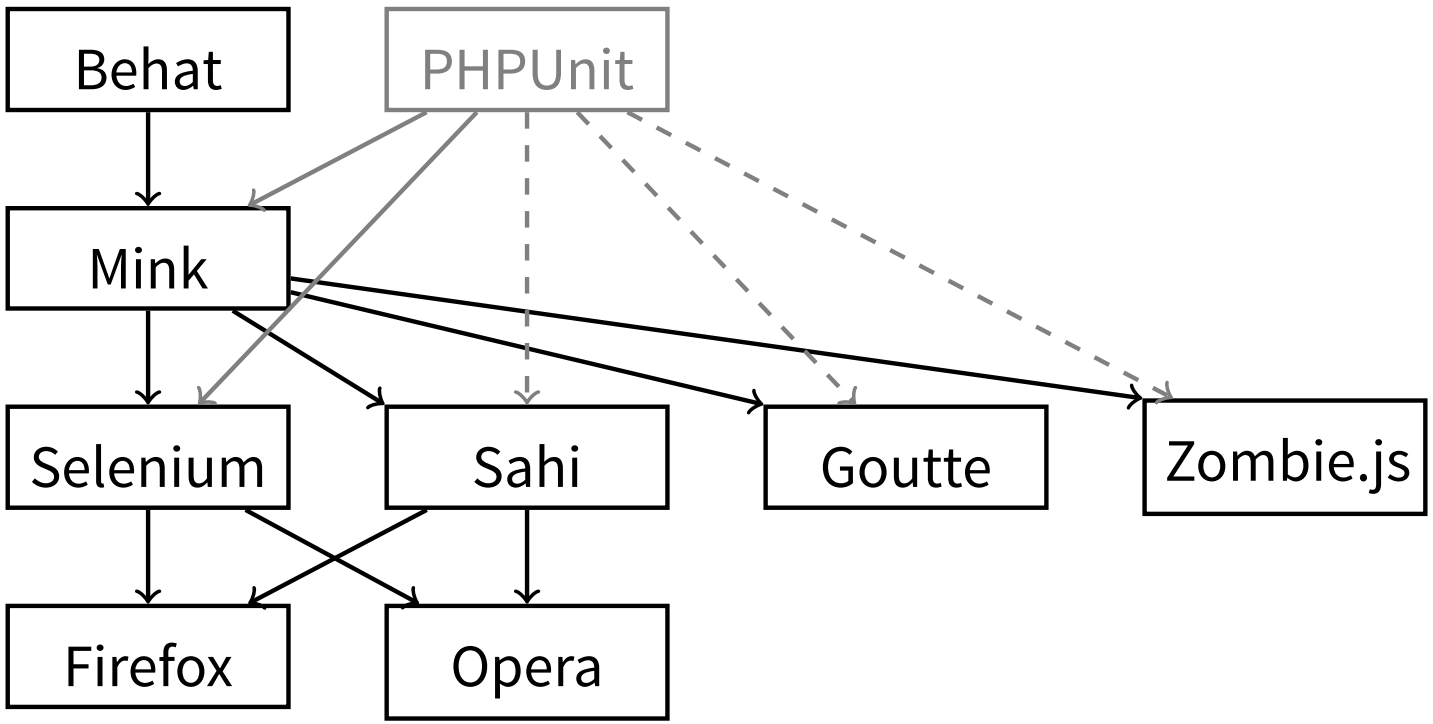
add method

```
/**
 * @Given /^ich suche nach "([^"]*)"$/
 */
public function ichSucheNach($arg)
{
    return array(
        new When("I fill in \"s\" with \"$arg\""),
        new When("I press \"searchsubmit\""),
    );
}
```

Behat Output

```
[mschuett@dagny] ~/blit/code% ./vendor/bin/behat
Funktionalität: Selbstpräsentation
  Um etwas über den BLIT zu erfahren
  Als Besucher
  Möchte ich Infos im Blog finden

  Szenario: Keysigning suchen                                  # features/blit.fea
    Angenommen ich bin auf ”/blog”                             # Behat\MinkExtensi
    Und ich suche nach ”keysigning”                            # FeatureContext::i
    Dann ich sollte ”Keysigning auf dem 9. BLIT” sehen # Behat\MinkExtensi

1 scenario (1 passed)
3 steps (3 passed)
0m1.439s
```

The Big Picture (PHP)



Problems

- most Web tests require known DB state
- behaviour testing is slow
  - only run locally
  - use in-memory DB
- draw the line
  - features in documentation (what) $\rightarrow$ add behaviour test
  - mere implementation (how) $\rightarrow$ only unit tests

Summary

Value individuals and interactions over processes and tools

– The Agile Manifesto

Links:

- PHPUnit
- Selenium
- Behat & Mink
- Codeception
- Cucumber
- Sebastian Bergmann (PHPUnit)
- Dan North's blog (BDD)
- Liz Keogh's blog (BDD)
- Geek and Poke
- Not Invented Here

Codeception unit test

a new full-stack testing PHP framework

```
$I = new CodeGuy($scenario);
$I->wantTo('perform actions and see results');

$I->testMethod('DateTime.getTimestamp');
$date = new DateTime("2012-10-13T14:45:00+02:00");
$I->executeTestedMethodOn($date);

$I->seeResultIs('int');
$I->seeResultEquals(1350132300);
```

Codeception acceptance test

```
$I = new WebGuy($scenario);
$I->wantTo('find infos on the website');

$I->amOnPage('/blog');
$I->see('BLIT-Blog', 'html head title');

$I->fillField('s', 'keysigning');
$I->click('searchsubmit');
$I->see('Keysigning auf dem 9. BLIT', '#content');
```

Codeception Output

```
[mschuett@dagny] ~/blit/code% php codecept.phar run
Codeception PHP Testing Framework v1.1.4
Powered by PHPUnit 3.6.10 by Sebastian Bergmann.

Suite unit started
Trying to perform actions and see results (exampleCept.php) - Ok

Suite functional started

Suite acceptance started
Trying to find infos on the website (blitCept.php) - Ok


Time: 1 second, Memory: 9.50Mb

OK (2 tests, 4 assertions)
```

Codeception Output

```
[mschuett@dagny] ~/blit/code% php codecept.phar run --steps
Codeception PHP Testing Framework v1.1.4
Powered by PHPUnit 3.6.10 by Sebastian Bergmann.

Suite unit started
Trying to perform actions and see results (exampleCept.php)
Scenario:
* I test method ”DateTime.getTimestamp”
* I execute tested method on ”DateTime”
* I see result is ”int”
* I see result equals ”1350132300”
  OK

Suite functional started

Suite acceptance started
Trying to find infos on the website (blitCept.php)
Scenario:
* I am on page ”/blog”
* I see ”BLIT-Blog”,”html head title”
* I fill field ”s”,”keysigning”
* I click ”searchsubmit”
* I see ”Keysigning auf dem 9. BLIT”,”#content”
  OK

Time: 1 second, Memory: 9.50Mb

OK (2 tests, 4 assertions)
```