Testen State of the Art

IPC Spring 2012

Manuel Pichler (@manuelp)
Tobias Schlitt (@tobySen)

2012-06-04

About us

- **Manuel** & **Toby**

About us

- **Manuel** & **Toby**
- Degree in computer sience
- More than 10 years of professional PHP

About us

- **Manuel** & **Toby**
- Degree in computer sience
- More than 10 years of professional PHP
- Open source enthusiasts
- Contributing to various FLOSS projects

About us

- **Manuel** & **Toby**
- Degree in computer sience
- More than 10 years of professional PHP
- Open source enthusiasts
- Contributing to various FLOSS projects

Co-founders of

About us

- **Manuel** & **Toby**
- Degree in computer sience
- More than 10 years of professional PHP
- Open source enthusiasts
- Contributing to various FLOSS projects

Co-founders of

Helping people to create high quality web applications.

About us

- **Manuel** & **Toby**
- Degree in computer sience
- More than 10 years of professional PHP
- Open source enthusiasts
- Contributing to various FLOSS projects

Co-founders of

Helping people to create high quality web applications.

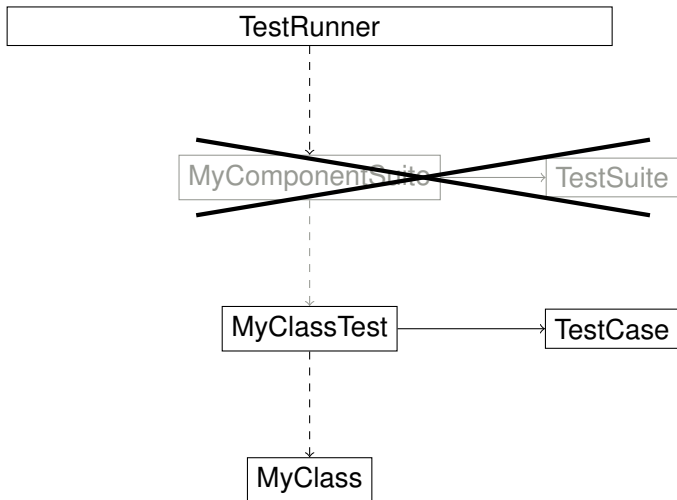`http://qafoo.com`

Outline

Unit Testing

PHPUnit schematics
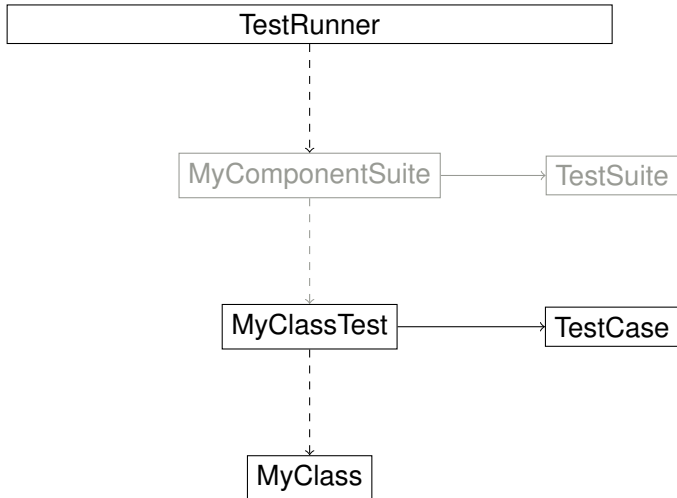
PHPUnit schematics

Test suites

- File system based
- phpunit.dist.xml / phpunit.xml
- @group annotations

phpunit.xml

- In the dir you execute `$ phpunit`
  - phpunit.xml available?
  - phpunit.dist.xml available?
- `$ phpunit -c my-phpunit.xml`

phpunit.xml

```xml
1  <phpunit bootstrap="libraries/autoload.php"
2           colors="true"
3           convertErrorsToExceptions="true"
4           convertNoticesToExceptions="true"
5           convertWarningsToExceptions="true"
6           >
7    <groups>
8      <exclude><group>integration</group></exclude>
9    </groups>
10
11   <testsuites>
12     <testsuite name="All_Tests">
13       <directory suffix=".php">tests/</directory>
14     </testsuite>
15   </testsuites>
16
17   <filter>
18     <whitelist>
19       <directory suffix=".php">../src/</directory>
20       <exclude>
21         <file>../src/bootstrap.php</file>
22       </exclude>
23     </whitelist>
24   </filter>
25 </phpunit>
```

phpunit.xml

```
1  <phpunit bootstrap="libraries/autoload.php"
2           colors="true"
3           convertErrorsToExceptions="true"
4           convertNoticesToExceptions="true"
5           convertWarningsToExceptions="true"
6           >
7    <groups>
8      <exclude><group>integration</group></exclude>
9    </groups>
10
11   <testsuites>
12     <testsuite name="All_Tests">
13       <directory suffix=".php">tests/</directory>
14     </testsuite>
15   </testsuites>
16
17   <filter>
18     <whitelist>
19       <directory suffix=".php">../src/</directory>
20       <exclude>
21         <file>../src/bootstrap.php</file>
22       </exclude>
23     </whitelist>
24   </filter>
25 </phpunit>
```

phpunit.xml

```xml
1  <phpunit bootstrap="libraries/autoload.php"
2           colors="true"
3           convertErrorsToExceptions="true"
4           convertNoticesToExceptions="true"
5           convertWarningsToExceptions="true"
6           >
7    <groups>
8      <exclude><group>integration</group></exclude>
9    </groups>
10
11   <testsuites>
12     <testsuite name="All_Tests">
13       <directory suffix=".php">tests/</directory>
14     </testsuite>
15   </testsuites>
16
17   <filter>
18     <whitelist>
19       <directory suffix=".php">../src/</directory>
20       <exclude>
21         <file>../src/bootstrap.php</file>
22       </exclude>
23     </whitelist>
24   </filter>
25 </phpunit>
```

phpunit.xml

```xml
<phpunit bootstrap="libraries/autoload.php"
         colors="true"
         convertErrorsToExceptions="true"
         convertNoticesToExceptions="true"
         convertWarningsToExceptions="true"
         >
  <groups>
    <exclude><group>integration</group></exclude>
  </groups>

  <testsuites>
    <testsuite name="All_Tests">
      <directory suffix=".php">tests/</directory>
    </testsuite>
  </testsuites>

  <filter>
    <whitelist>
      <directory suffix=".php">../src/</directory>
      <exclude>
        <file>../src/bootstrap.php</file>
      </exclude>
    </whitelist>
  </filter>
</phpunit>
```

Useful Annotations

- `@covers`

Useful Annotations

- @covers
- @group
- @ticket

Useful Annotations

- @covers
- @group
- @ticket
- **@dataProvider**

Useful Annotations

- ▶ @covers
- ▶ @group
- ▶ @ticket
- ▶ @dataProvider
- ▶ **@expectedException***

Useful Annotations

- ► @covers
- ► @group
- ► @ticket
- ► @dataProvider
- ► @expectedException*
- ► @codeCoverageIgnore*

Useful Annotations

- ▶ @covers
- ▶ @group
- ▶ @ticket
- ▶ @dataProvider
- ▶ @expectedException*
- ▶ @codeCoverageIgnore*
- ▶ @depends

Outline

Unit Testing

Testing File Uploads

Behavior Driven Development

Automated Acceptance Tests

Testing JavaScript

Outlook

A Minimal File Upload

```php
<?php
class UploadExample
{
    protected $dest;
    public function __construct($dest)
    {
        $this->dest = rtrim($dest, '/') . '/';
    }
    public function handle($name)
    {
        if (is_uploaded_file($_FILES[$name]['tmp_name'])) {
            move_uploaded_file(
                $_FILES[$name]['tmp_name'],
                $this->dest . $_FILES[$name]['name']
            );
        }
    }
}
```

A Minimal File Upload

```php
<?php
class UploadExample
{
    protected $dest;
    public function __construct($dest)
    {
        $this->dest = rtrim($dest, '/') . '/';
    }
    public function handle($name)
    {
        if (is_uploaded_file($_FILES[$name]['tmp_name'])) {
            move_uploaded_file(
                $_FILES[$name]['tmp_name'],
                $this->dest . $_FILES[$name]['name']
            );
        }
    }
}
```

PHPT

- ▶ PHP Testing Framework (PHPT)
- ▶ Used to test PHP core
- ▶ $ pear run-tests example-test-case.phpt
- ▶ Integrates into PHPUnit
- ▶ Details: http://qa.php.net/phpt_details.php

A Simple PHPT Test

```
1  --TEST--
2  Example test case
3  --FILE--
4  <?php
5  var_dump(strpos('Manuel Pichler', 'P'));
6  var_dump(strpos('Manuel Pichler', 'Z'));
7  --EXPECT--
8  int(7)
9  bool(false)
```

A Simple PHPT Test

```
1   −−TEST−−
2   Example test case
3   −−FILE−−
4   <?php
5   var_dump(strpos('Manuel Pichler', 'P'));
6   var_dump(strpos('Manuel Pichler', 'Z'));
7   −−EXPECT−−
8   int(7)
9   bool(false)
```

A Simple PHPT Test

```
--TEST--
Example test case
--FILE--
<?php
var_dump(strpos('Manuel Pichler', 'P'));
var_dump(strpos('Manuel Pichler', 'Z'));
--EXPECT--
int(7)
bool(false)
```

File Uploads with PHPT

```
1    --TEST--
2    Example test emulating a file upload
3    --POST_RAW--
4    Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryfywL8UCjFtqUBTQn
5
6    ------WebKitFormBoundaryfywL8UCjFtqUBTQn
7    Content-Disposition: form-data; name="file"; filename="example.txt"
8    Content-Type: text/plain
9
10   Qafoo provides quality assurance support and consulting
11
12   ------WebKitFormBoundaryfywL8UCjFtqUBTQn
13   Content-Disposition: form-data; name="submit"
14
15   Upload
16   ------WebKitFormBoundaryfywL8UCjFtqUBTQn--
17   --FILE--
18   <?php
19   require __DIR__ . '/UploadExample.php';
20
21   $upload = new UploadExample('/tmp');
22   $upload->handle('file');
23
24   var_dump(file_exists('/tmp/example.txt'));
25   ?>
26   --EXPECT--
27   bool(true)
```

File Uploads with PHPT

```
1    ‑‑TEST‑‑
2    Example test emulating a file upload
3    ‑‑POST_RAW‑‑
4    Content‑Type: multipart/form‑data; boundary=‑‑‑‑WebKitFormBoundaryfywL8UCjFtqUBTQn
5
6    ‑‑‑‑‑‑WebKitFormBoundaryfywL8UCjFtqUBTQn
7    Content‑Disposition: form‑data; name="file"; filename="example.txt"
8    Content‑Type: text/plain
9
10   Qafoo provides quality assurance support and consulting
11
12   ‑‑‑‑‑‑WebKitFormBoundaryfywL8UCjFtqUBTQn
13   Content‑Disposition: form‑data; name="submit"
14
15   Upload
16   ‑‑‑‑‑‑WebKitFormBoundaryfywL8UCjFtqUBTQn‑‑
17   ‑‑FILE‑‑
18   <?php
19   require __DIR__ . '/UploadExample.php';
20
21   $upload = new UploadExample('/tmp');
22   $upload->handle('file');
23
24   var_dump(file_exists('/tmp/example.txt'));
25   ?>
26   ‑‑EXPECT‑‑
27   bool(true)
```

File Uploads with PHPT

```
 1   ––TEST––
 2   Example test emulating a file upload
 3   ––POST_RAW––
 4   Content-Type: multipart/form-data; boundary=––––WebKitFormBoundaryfywL8UCjFtqUBTQn
 5
 6   ––––––WebKitFormBoundaryfywL8UCjFtqUBTQn
 7   Content-Disposition: form-data; name="file"; filename="example.txt"
 8   Content-Type: text/plain
 9
10   Qafoo provides quality assurance support and consulting
11
12   ––––––WebKitFormBoundaryfywL8UCjFtqUBTQn
13   Content-Disposition: form-data; name="submit"
14
15   Upload
16   ––––––WebKitFormBoundaryfywL8UCjFtqUBTQn––
17   ––FILE––
18   <?php
19   require __DIR__ . '/UploadExample.php';
20
21   $upload = new UploadExample('/tmp');
22   $upload->handle('file');
23
24   var_dump(file_exists('/tmp/example.txt'));
25   ?>
26   ––EXPECT––
27   bool(true)
```

File Uploads with PHPT

```
1    −−TEST−−
2    Example test emulating a file upload
3    −−POST_RAW−−
4    Content−Type: multipart/form−data; boundary=−−−−WebKitFormBoundaryfywL8UCjFtqUBTQn
5
6    −−−−−−WebKitFormBoundaryfywL8UCjFtqUBTQn
7    Content−Disposition: form−data; name="file"; filename="example.txt"
8    Content−Type: text/plain
9
10   Qafoo provides quality assurance support and consulting
11
12   −−−−−−WebKitFormBoundaryfywL8UCjFtqUBTQn
13   Content−Disposition: form−data; name="submit"
14
15   Upload
16   −−−−−−WebKitFormBoundaryfywL8UCjFtqUBTQn−−
17   −−FILE−−
18   <?php
19   require __DIR__ . '/UploadExample.php';
20
21   $upload = new UploadExample('/tmp');
22   $upload−>handle('file');
23
24   var_dump(file_exists('/tmp/example.txt'));
25   ?>
26   −−EXPECT−−
27   bool(true)
```

PHPT within PHPUnit

```
1  <phpunit>
2    <testsuites>
3      <testsuite name="PHPT">
4          <directory suffix=".phpt">./</directory>
5      </testsuite>
6    </testsuites>
7  </phpunit>
```

Outline

Unit Testing

Testing File Uploads

Behavior Driven Development

Automated Acceptance Tests

Testing JavaScript

Outlook

Behavior Driven Development

- ► Behavior Driven Development (BDD)
- ► Deeper coupling between
  - ► Business Analysis
  - ► Quality Assurance

Behavior Driven Development

- ▶ Behavior Driven Development (BDD)
- ▶ Deeper coupling between
  - ▶ Business Analysis
  - ▶ Quality Assurance
- ▶ Basic ideas:
  - ▶ Use cases as requirements
  - ▶ Make use cases executable
  - ▶ Verify functionality

Behavior Driven Development

- ▶ Behavior Driven Development (BDD)
- ▶ Deeper coupling between
  - ▶ Business Analysis
  - ▶ Quality Assurance
- ▶ Basic ideas:
  - ▶ Use cases as requirements
  - ▶ Make use cases executable
  - ▶ Verify functionality
- ▶ Advantages:
  - ▶ Communication
  - ▶ Documentation

Behavior Driven Development

- ▶ Behavior Driven Development (BDD)
- ▶ Deeper coupling between
  - ▶ Business Analysis
  - ▶ Quality Assurance
- ▶ Basic ideas:
  - ▶ Use cases as requirements
  - ▶ Make use cases executable
  - ▶ Verify functionality
- ▶ Advantages:
  - ▶ Communication
  - ▶ Documentation
- ▶ Short **Human readable tests**

The BDD process

1. Define a Domain Specific Language (DSL)

2. Describe Use Cases

3. Implement against Use Case execution

Behat

- BDD framework for PHP
- http://behat.org/
- Inspired by Ruby's Cucumber
- Getting started:
  http://docs.behat.org/quick_intro.html

Defining Features

```
1   Feature:
2     BDD tests for the eZ Publish Section API
3
4     Scenario: Admin user can delete a section
5       Given I have a section with name "MySection" and identifier "MySectionKey"
6         And I am logged in as user "admin"
7       When I delete section with identifier "MySectionKey"
8       Then I expect no section for identifier "MySectionKey" exists
9
10    Scenario: Anonymous user cannot delete a section
11      Given I have a section with name "MySection" and identifier "MySectionKey"
12        And I am logged in as user "anonymous"
13      When I delete section with identifier "MySectionKey"
14      Then I expect delete failes with an exception of type "unauthorized"
```

Defining Features

```
1   Feature:
2     BDD tests for the eZ Publish Section API
3
4     Scenario: Admin user can delete a section
5       Given I have a section with name "MySection" and identifier "MySectionKey"
6         And I am logged in as user "admin"
7       When I delete section with identifier "MySectionKey"
8       Then I expect no section for identifier "MySectionKey" exists
9
10    Scenario: Anonymous user cannot delete a section
11      Given I have a section with name "MySection" and identifier "MySectionKey"
12        And I am logged in as user "anonymous"
13      When I delete section with identifier "MySectionKey"
14      Then I expect delete failes with an exception of type "unauthorized"
```

Defining Features

```
1   Feature:
2     BDD tests for the eZ Publish Section API
3
4     Scenario: Admin user can delete a section
5       Given I have a section with name "MySection" and identifier "MySectionKey"
6         And I am logged in as user "admin"
7       When I delete section with identifier "MySectionKey"
8       Then I expect no section for identifier "MySectionKey" exists
9
10    Scenario: Anonymous user cannot delete a section
11      Given I have a section with name "MySection" and identifier "MySectionKey"
12        And I am logged in as user "anonymous"
13      When I delete section with identifier "MySectionKey"
14      Then I expect delete failes with an exception of type "unauthorized"
```

Defining Features

```
1    Feature:
2      BDD tests for the eZ Publish Section API
3
4      Scenario: Admin user can delete a section
5        Given I have a section with name "MySection" and identifier "MySectionKey"
6          And I am logged in as user "admin"
7        When I delete section with identifier "MySectionKey"
8        Then I expect no section for identifier "MySectionKey" exists
9
10     Scenario: Anonymous user cannot delete a section
11       Given I have a section with name "MySection" and identifier "MySectionKey"
12         And I am logged in as user "anonymous"
13       When I delete section with identifier "MySectionKey"
14       Then I expect delete failes with an exception of type "unauthorized"
```

Defining Features

```
1   Feature:
2     BDD tests for the eZ Publish Section API
3
4     Scenario: Admin user can delete a section
5       Given I have a section with name "MySection" and identifier "MySectionKey"
6         And I am logged in as user "admin"
7       When I delete section with identifier "MySectionKey"
8       Then I expect no section for identifier "MySectionKey" exists
9
10    Scenario: Anonymous user cannot delete a section
11      Given I have a section with name "MySection" and identifier "MySectionKey"
12        And I am logged in as user "anonymous"
13      When I delete section with identifier "MySectionKey"
14      Then I expect delete failes with an exception of type "unauthorized"
```

Defining Features

```
1   Feature:
2     BDD tests for the eZ Publish Section API
3
4     Scenario: Admin user can delete a section
5       Given I have a section with name "MySection" and identifier "MySectionKey"
6         And I am logged in as user "admin"
7        When I delete section with identifier "MySectionKey"
8        Then I expect no section for identifier "MySectionKey" exists
9
10    Scenario: Anonymous user cannot delete a section
11      Given I have a section with name "MySection" and identifier "MySectionKey"
12        And I am logged in as user "anonymous"
13       When I delete section with identifier "MySectionKey"
14       Then I expect delete failes with an exception of type "unauthorized"
```

Defining Features

```
1   Feature:
2     BDD tests for the eZ Publish Section API
3
4     Scenario: Admin user can delete a section
5       Given I have a section with name "MySection" and identifier "MySectionKey"
6         And I am logged in as user "admin"
7       When I delete section with identifier "MySectionKey"
8       Then I expect no section for identifier "MySectionKey" exists
9
10    Scenario: Anonymous user cannot delete a section
11      Given I have a section with name "MySection" and identifier "MySectionKey"
12        And I am logged in as user "anonymous"
13      When I delete section with identifier "MySectionKey"
14      Then I expect delete failes with an exception of type "unauthorized"
```

Implementing the DSL

```
16   class FeatureContext extends BehatContext
17   {

48       /**
49        * @Given /^I have a section with name "([^"]+)" and identifier "([^"]+)"$/
50        */
51       public function iHaveASectionWithNameAndIdentifier( $name, $identifier )
52       {
53           $sectionService = $this->repository->getSectionService();
54
55           $sectionCreate            = $sectionService->newSectionCreateStruct();
56           $sectionCreate->name      = $name;
57           $sectionCreate->identifier = $identifier;
58
59           $this->runtimeData[ $identifier ] = $sectionService->createSection( $sectionCreate
                 );
60       }

112  }
```

Implementing the DSL

```
16   class FeatureContext extends BehatContext
17   {

48       /**
49        * @Given /^I have a section with name "([^"]+)" and identifier "([^"]+)"$/
50        */
51       public function iHaveASectionWithNameAndIdentifier( $name, $identifier )
52       {
53           $sectionService = $this->repository->getSectionService();
54
55           $sectionCreate             = $sectionService->newSectionCreateStruct();
56           $sectionCreate->name       = $name;
57           $sectionCreate->identifier = $identifier;
58
59           $this->runtimeData[$identifier] = $sectionService->createSection( $sectionCreate
                 );
60       }

112  }
```

Implementing the DSL

```
16   class FeatureContext extends BehatContext
17   {

48       /**
49        * @Given /^I have a section with name "([^"]+)" and identifier "([^"]+)"$/
50        */
51       public function iHaveASectionWithNameAndIdentifier( $name, $identifier )
52       {
53           $sectionService = $this->repository->getSectionService();
54
55           $sectionCreate             = $sectionService->newSectionCreateStruct();
56           $sectionCreate->name       = $name;
57           $sectionCreate->identifier = $identifier;
58
59           $this->runtimeData[$identifier] = $sectionService->createSection( $sectionCreate
                    );
60       }

112  }
```

Implementing the DSL

```
16   class FeatureContext extends BehatContext
17   {

48       /**
49        * @Given /^I have a section with name "([^"]+)" and identifier "([^"]+)"$/
50        */
51       public function iHaveASectionWithNameAndIdentifier( $name, $identifier )
52       {
53           $sectionService = $this->repository->getSectionService();
54
55           $sectionCreate             = $sectionService->newSectionCreateStruct();
56           $sectionCreate->name       = $name;
57           $sectionCreate->identifier = $identifier;
58
59           $this->runtimeData[$identifier] = $sectionService->createSection( $sectionCreate
                    );
60       }

112  }
```

Implementing the DSL

```php
16    class FeatureContext extends BehatContext
17    {

48        /**
49         * @Given /^I have a section with name "([^"]+)" and identifier "([^"]+)"$/
50         */
51        public function iHaveASectionWithNameAndIdentifier( $name, $identifier )
52        {
53            $sectionService = $this->repository->getSectionService();
54
55            $sectionCreate             = $sectionService->newSectionCreateStruct();
56            $sectionCreate->name       = $name;
57            $sectionCreate->identifier = $identifier;
58
59            $this->runtimeData[$identifier] = $sectionService->createSection( $sectionCreate
                   );
60        }

112   }
```

Implementing the DSL

```
16   class FeatureContext extends BehatContext
17   {

48       /**
49        * @Given /^I have a section with name "([^"]+)" and identifier "([^"]+)"$/
50        */
51       public function iHaveASectionWithNameAndIdentifier( $name, $identifier )
52       {
53           $sectionService = $this->repository->getSectionService();
54
55           $sectionCreate             = $sectionService->newSectionCreateStruct();
56           $sectionCreate->name       = $name;
57           $sectionCreate->identifier = $identifier;
58
59           $this->runtimeData[$identifier] = $sectionService->createSection( $sectionCreate
                  );
60       }

112  }
```

When . . .

```
1    Feature:
2      BDD tests for the eZ Publish Section API
3
4      Scenario: Admin user can delete a section
5        Given I have a section with name "MySection" and identifier "MySectionKey"
6          And I am logged in as user "admin"
7        When I delete section with identifier "MySectionKey"
8        Then I expect no section for identifier "MySectionKey" exists
9
10     Scenario: Anonymous user cannot delete a section
11       Given I have a section with name "MySection" and identifier "MySectionKey"
12         And I am logged in as user "anonymous"
13       When I delete section with identifier "MySectionKey"
14       Then I expect delete fails with an exception of type "unauthorized"
```

Implementing the DSL

```php
class FeatureContext extends BehatContext
{


    /**
     * @When /^I delete section with identifier "([^"]+)"$/
     */
    public function iDeleteSectionWithIdentifier( $identifier )
    {
        $sectionService = $this->repository->getSectionService();

        try
        {
            $section = $sectionService->loadSectionByIdentifier( $identifier );
            $sectionService->deleteSection( $section );
        }
        catch ( \Exception $e )
        {
            $this->runtimeData['__exception'] = $e;
        }
    }


}
```

Implementing the DSL

```php
class FeatureContext extends BehatContext
{

    /**
     * @When /^I delete section with identifier "([^"]+)"$/
     */
    public function iDeleteSectionWithIdentifier( $identifier )
    {
        $sectionService = $this->repository->getSectionService();

        try
        {
            $section = $sectionService->loadSectionByIdentifier( $identifier );
            $sectionService->deleteSection( $section );
        }
        catch ( \Exception $e )
        {
            $this->runtimeData['__exception'] = $e;
        }
    }


}
```

Then ...

```
1   Feature:
2     BDD tests for the eZ Publish Section API
3
4     Scenario: Admin user can delete a section
5       Given I have a section with name "MySection" and identifier "MySectionKey"
6         And I am logged in as user "admin"
7        When I delete section with identifier "MySectionKey"
8        Then I expect no section for identifier "MySectionKey" exists
9
10    Scenario: Anonymous user cannot delete a section
11      Given I have a section with name "MySection" and identifier "MySectionKey"
12        And I am logged in as user "anonymous"
13       When I delete section with identifier "MySectionKey"
14       Then I expect delete failes with an exception of type "unauthorized"
```

Implementing the DSL

```php
class FeatureContext extends BehatContext
{


    /**
     * @Then /^I expect no section for identifier "([^"]+)" exists$/
     */
    public function iExpectNoSectionForIdentifierExists( $identifier )
    {
        $sectionService = $this->repository->getSectionService();
        try
        {
            $sectionService->loadSectionByIdentifier( $identifier );
            throw new \RuntimeException( "Identifier ($identifier) still exists." );
        }
        catch ( \eZ\Publish\API\Repository\Exceptions\NotFoundException $e )
        {
        }
    }


}
```

Implementing the DSL

```
16   class FeatureContext extends BehatContext
17   {

87       /**
88        * @Then /^I expect no section for identifier "([^"]+)" exists$/
89        */
90       public function iExpectNoSectionForIdentifierExists( $identifier )
91       {
92           $sectionService = $this->repository->getSectionService();
93           try
94           {
95               $sectionService->loadSectionByIdentifier( $identifier );
96               throw new \RuntimeException( "Identifier [$identifier] still exists." );
97           }
98           catch ( \eZ\Publish\API\Repository\Exceptions\NotFoundException $e )
99           {
100          }
101      }

112  }
```

Outline

Unit Testing

Testing File Uploads

Behavior Driven Development

Automated Acceptance Tests

Testing JavaScript

Outlook

Automated Acceptance Tests

- ▶ High level system tests
- ▶ Assert user expectations
- ▶ Good start for refactoring
- ▶ Approaches
  - ▶ HTTP spider (e.g. Goutte)
  - ▶ Browser controller (e.g. Selenium, Sahi)
  - ▶ Browser emulator (e.g. Zombie.js)

Mink

- Acceptance Test Framework
- `http://mink.behat.org`
- Abstraction for underlying tools
- Full integration into Behat

Testing a Registration Form

```
1   Feature: Registration
2     In order to perform actions like an order
3     As a normal user
4     I must perform a registration
5
6     Scenario: Test a default registration
7       Given I am on "/register"
8       When I select "private" from "register_personal_customer_type"
9       And I select "mr" from "register_personal_salutation_mr"
10      And I fill in "firstname" with "Tobias"
11      And I fill in "lastname" with "Schlitt"
12      And I fill in "register_personal_email" with "ts01@qafoo.com"
13      And I fill in "register_personal_password" with "sindelfingen"
14      And I fill in "register_personal_passwordConfirmation" with "sindelfingen"
15      And I fill in "phone" with "+49 (0)209 40501252"
16      And I fill in "street" with "Bochumer Strasse"
17      And I fill in "streetnumber" with "226"
18      And I fill in "zipcode" with "45886"
19      And I fill in "city" with "Gelsenkirchen"
20      And I select "2" from "country"
21      And I press "registerbutton"
22      Then I should see "Willkommen, Tobias Schlitt"
```

Testing a Registration Form

```
1   Feature: Registration
2     In order to perform actions like an order
3     As a normal user
4     I must perform a registration
5
6     Scenario: Test a default registration
7       Given I am on "/register"
8       When I select "private" from "register_personal_customer_type"
9       And I select "mr" from "register_personal_salutation_mr"
10      And I fill in "firstname" with "Tobias"
11      And I fill in "lastname" with "Schlitt"
12      And I fill in "register_personal_email" with "ts01@qafoo.com"
13      And I fill in "register_personal_password" with "sindelfingen"
14      And I fill in "register_personal_passwordConfirmation" with "sindelfingen"
15      And I fill in "phone" with "+49 (0)209 40501252"
16      And I fill in "street" with "Bochumer Strasse"
17      And I fill in "streetnumber" with "226"
18      And I fill in "zipcode" with "45886"
19      And I fill in "city" with "Gelsenkirchen"
20      And I select "2" from "country"
21      And I press "registerbutton"
22      Then I should see "Willkommen, Tobias Schlitt"
```

Testing a Registration Form

```
1   Feature: Registration
2     In order to perform actions like an order
3     As a normal user
4     I must perform a registration
5
6     Scenario: Test a default registration
7       Given I am on "/register"
8       When I select "private" from "register_personal_customer_type"
9       And I select "mr" from "register_personal_salutation_mr"
10      And I fill in "firstname" with "Tobias"
11      And I fill in "lastname" with "Schlitt"
12      And I fill in "register_personal_email" with "ts01@qafoo.com"
13      And I fill in "register_personal_password" with "sindelfingen"
14      And I fill in "register_personal_passwordConfirmation" with "sindelfingen"
15      And I fill in "phone" with "+49 (0)209 40501252"
16      And I fill in "street" with "Bochumer Strasse"
17      And I fill in "streetnumber" with "226"
18      And I fill in "zipcode" with "45886"
19      And I fill in "city" with "Gelsenkirchen"
20      And I select "2" from "country"
21      And I press "registerbutton"
22      Then I should see "Willkommen, Tobias Schlitt"
```

Testing a Registration Form

```
1   Feature: Registration
2     In order to perform actions like an order
3     As a normal user
4     I must perform a registration
5
6     Scenario: Test a default registration
7       Given I am on "/register"
8       When I select "private" from "register_personal_customer_type"
9       And I select "mr" from "register_personal_salutation_mr"
10      And I fill in "firstname" with "Tobias"
11      And I fill in "lastname" with "Schlitt"
12      And I fill in "register_personal_email" with "ts01@qafoo.com"
13      And I fill in "register_personal_password" with "sindelfingen"
14      And I fill in "register_personal_passwordConfirmation" with "sindelfingen"
15      And I fill in "phone" with "+49 (0)209 40501252"
16      And I fill in "street" with "Bochumer Strasse"
17      And I fill in "streetnumber" with "226"
18      And I fill in "zipcode" with "45886"
19      And I fill in "city" with "Gelsenkirchen"
20      And I select "2" from "country"
21      And I press "registerbutton"
22      Then I should see "Willkommen, Tobias Schlitt"
```

Testing a Registration Form

```
 1   Feature: Registration
 2     In order to perform actions like an order
 3     As a normal user
 4     I must perform a registration
 5
 6     Scenario: Test a default registration
 7       Given I am on "/register"
 8       When I select "private" from "register_personal_customer_type"
 9       And I select "mr" from "register_personal_salutation_mr"
10       And I fill in "firstname" with "Tobias"
11       And I fill in "lastname" with "Schlitt"
12       And I fill in "register_personal_email" with "ts01@qafoo.com"
13       And I fill in "register_personal_password" with "sindelfingen"
14       And I fill in "register_personal_passwordConfirmation" with "sindelfingen"
15       And I fill in "phone" with "+49 (0)209 40501252"
16       And I fill in "street" with "Bochumer Strasse"
17       And I fill in "streetnumber" with "226"
18       And I fill in "zipcode" with "45886"
19       And I fill in "city" with "Gelsenkirchen"
20       And I select "2" from "country"
21       And I press "registerbutton"
22       Then I should see "Willkommen, Tobias Schlitt"
```

Testing a Registration Form

```
1   Feature: Registration
2     In order to perform actions like an order
3     As a normal user
4     I must perform a registration
5
6     Scenario: Test a default registration
7       Given I am on "/register"
8       When I select "private" from "register_personal_customer_type"
9       And I select "mr" from "register_personal_salutation_mr"
10      And I fill in "firstname" with "Tobias"
11      And I fill in "lastname" with "Schlitt"
12      And I fill in "register_personal_email" with "ts01@qafoo.com"
13      And I fill in "register_personal_password" with "sindelfingen"
14      And I fill in "register_personal_passwordConfirmation" with "sindelfingen"
15      And I fill in "phone" with "+49 (0)209 40501252"
16      And I fill in "street" with "Bochumer Strasse"
17      And I fill in "streetnumber" with "226"
18      And I fill in "zipcode" with "45886"
19      And I fill in "city" with "Gelsenkirchen"
20      And I select "2" from "country"
21      And I press "registerbutton"
22      Then I should see "Willkommen, Tobias Schlitt"
```

Testing a Registration Form

```
1   Feature: Registration
2     In order to perform actions like an order
3     As a normal user
4     I must perform a registration
5
6     Scenario: Test a default registration
7       Given I am on "/register"
8       When I select "private" from "register_personal_customer_type"
9       And I select "mr" from "register_personal_salutation_mr"
10      And I fill in "firstname" with "Tobias"
11      And I fill in "lastname" with "Schlitt"
12      And I fill in "register_personal_email" with "ts01@qafoo.com"
13      And I fill in "register_personal_password" with "sindelfingen"
14      And I fill in "register_personal_passwordConfirmation" with "sindelfingen"
15      And I fill in "phone" with "+49 (0)209 40501252"
16      And I fill in "street" with "Bochumer Strasse"
17      And I fill in "streetnumber" with "226"
18      And I fill in "zipcode" with "45886"
19      And I fill in "city" with "Gelsenkirchen"
20      And I select "2" from "country"
21      And I press "registerbutton"
22      Then I should see "Willkommen, Tobias Schlitt"
```

Outline

Unit Testing

Testing File Uploads

Behavior Driven Development

Automated Acceptance Tests

Testing JavaScript

Outlook

Testing JavaScript

- ▶ You test PHP . . .
- ▶ . . . do you test JavaScript?

Qafoo
passion for software quality

js-test-driver

- ▶ Test runner for JavaScript
- ▶ `https://code.google.com/p/js-test-driver/`
- ▶ Test JS inside browsers
- ▶ Run JS tests from command line
- ▶ No HTML required

Simple JS Code

```
1  var Hello = function() {};
2  Hello.prototype = {
3      say: function( who ) {
4          if ( who === undefined ) {
5              who = "World";
6          }
7          return "Hello " + who + "!";
8      }
9  }
```

Simple JS Code

```
1  var Hello = function() {};
2  Hello.prototype = {
3      say: function( who ) {
4          if ( who === undefined ) {
5              who = "World";
6          }
7          return "Hello " + who + "!";
8      }
9  }
```

Simple JS Code

```js
var Hello = function () {};
Hello.prototype = {
    say: function ( who ) {
        if ( who === undefined ) {
            who = "World";
        }
        return "Hello " + who + "!";
    }
}
```

Simple JS Code

```
1  var Hello = function() {};
2  Hello.prototype = {
3      say: function( who ) {
4          if ( who === undefined ) {
5              who = "World";
6          }
7          return "Hello " + who + "!";
8      }
9  }
```

A Simple Test Case

```
TestCase( "Hello object tests", {
  "setUp": function () {
    this.hello = new Hello();
  },
  "testCallableWithoutArgument": function() {
    assertEquals( "Hello World!", this.hello.say() );
  },
  "testCallableWithArgument": function() {
    assertEquals( "Hello Jakob!", this.hello.say( "Jakob" ) );
  }
});
```

A Simple Test Case

```
TestCase( "Hello object tests", {
  "setUp": function () {
    this.hello = new Hello();
  },
  "testCallableWithoutArgument": function() {
    assertEquals( "Hello World!", this.hello.say() );
  },
  "testCallableWithArgument": function() {
    assertEquals( "Hello Jakob!", this.hello.say( "Jakob" ) );
  }
} );
```

A Simple Test Case

```
TestCase( "Hello_object_tests", {
  "setUp": function () {
    this.hello = new Hello();
  },
  "testCallableWithoutArgument": function () {
    assertEquals( "Hello_World!", this.hello.say() );
  },
  "testCallableWithArgument": function () {
    assertEquals( "Hello_Jakob!", this.hello.say( "Jakob" ) );
  }
});
```

A Simple Test Case

```
TestCase( "Hello_object_tests", {
  "setUp": function () {
    this.hello = new Hello();
  },
  "testCallableWithoutArgument": function() {
    assertEquals( "Hello_World!", this.hello.say() );
  },
  "testCallableWithArgument": function() {
    assertEquals( "Hello_Jakob!", this.hello.say( "Jakob" ) );
  }
});
```

Configuring js-test-driver

```
1  server: http://localhost:4223
2
3  load:
4    - src/hello.js
5    - tests/hello.js
```

Configuring js-test-driver

```
1    server: http://localhost:4223
2
3    load:
4      - src/hello.js
5      - tests/hello.js
```

Running the Tests

- ▶ First shell

```
1   $ java -jar JsTestDriver-1.3.4.b.jar --port 4223
```

Running the Tests

- ▶ First shell

```
1   $ java −jar JsTestDriver−1.3.4.b.jar −−port 4223
```

- ▶ Capture browser
  - ▶ http://localhost:4223/capture

Running the Tests

- ▶ **First shell**

```
1   $ java −jar JsTestDriver −1.3.4.b.jar −−port 4223
```

- ▶ **Capture browser**
  - ▶ `http://localhost:4223/capture`

- ▶ **Second shell**

```
1   $ java −jar JsTestDriver −1.3.4.b.jar −−tests all
2   setting runnermode QUIET
3   ....
4   Total 4 tests (Passed: 4; Fails: 0; Errors: 0) (1.00 ms)
5   Firefox 12.0 Linux: Run 2 tests (Passed: 2; Fails: 0; Errors 0) (1.00 ms)
6   Chrome 18.0.1025.168 Linux: Run 2 tests (Passed: 2; Fails: 0; Errors 0) (0.00 ms)
```

JavaScript QA: js-test-driver und Sinon.JS richtig nutzen

- ▸ Jakob Westhoff (Qafoo)
- ▸ *Wednesday, 14:00, Salon 2*

Outline

Unified Build Process

- `https://github.com/manuelpichler/build-commons`
- Framework for build scripts

Unified Build Process

- `https://github.com/manuelpichler/build-commons`
- Framework for build scripts
- Unified targets, like
  - test
  - package
  - verify
  - deploy

Unified Build Process

- `https://github.com/manuelpichler/build-commons`
- Framework for build scripts
- Unified targets, like
    - test
    - package
    - verify
    - deploy
- **39 supported tool extensions**
    - PHPUnit, CodeSniffer, PDepend, PHPMD, PHPCPD, . . .
    - DB, DBDeploy, PHAR, Pirum, . . .
    - js-test-driver, JSLint, JSHint, . . .
    - . . .

Continuous Performance

- ▶ Does your app still perform?
  - ▶ Find performance regressions
  - ▶ Ensure optimizations persist

Continuous Performance

- ▶ Does your app still perform?
    - ▶ Find performance regressions
    - ▶ Ensure optimizations persist
- ▶ Apache JMeter
    - ▶ https://jmeter.apache.org/
    - ▶ Complex user paths
    - ▶ Concurrend requests
    - ▶ Record on proxy
    - ▶ Clustering

Continuous Performance

- ▶ Does your app still perform?
  - ▶ Find performance regressions
  - ▶ Ensure optimizations persist
- ▶ Apache JMeter
  - ▶ https://jmeter.apache.org/
  - ▶ Complex user paths
  - ▶ Concurrend requests
  - ▶ Record on proxy
  - ▶ Clustering
- ▶ Integrate into Continuous Integration

Continuous Performance

- ▶ Does your app still perform?
  - ▸ Find performance regressions
  - ▸ Ensure optimizations persist
- ▶ Apache JMeter
  - ▸ https://jmeter.apache.org/
  - ▸ Complex user paths
  - ▸ Concurrend requests
  - ▸ Record on proxy
  - ▸ Clustering
- ▶ Integrate into Continuous Integration

Immer am Optimum, Continuous Performance

- ▶ Wednesday, 16:15, Salon 2
- ▶ Kore Nordmann, Manuel Pichler

Thanks for Listening

Please rate this talk at
`https://joind.in/event/IPC2012`
(Slides: `http://talks.qafoo.com`)

Thanks for Listening

Please rate this talk at
`https://joind.in/event/IPC2012`
(Slides: `http://talks.qafoo.com`)

Stay in touch

- Manuel Pichler
- manuel@qafoo.com
- @manuelp / @qafoo

- Tobias Schlitt
- toby@qafoo.com
- @tobySen / @qafoo

Please rate this talk at
`https://joind.in/event/IPC2012`
(Slides: `http://talks.qafoo.com`)

Stay in touch

- Manuel Pichler
- manuel@qafoo.com
- @manuelp / @qafoo

- Tobias Schlitt
- toby@qafoo.com
- @tobySen / @qafoo

Rent a PHP quality expert:
`http://qafoo.com`