A generic PHP Application installer

PHPDay Italy 2011

Kore Nordmann & Tobias Schlitt

May 12, 2011

Qafoo
passion for software quality

About me

- ▶ Degree in computer sience in 2010
- ▶ More than 10 years of professional PHP
- ▶ Open source enthusiast
- ▶ Contributing to various FLOSS projects
- ▶ Founder of **Qafoo GmbH**, which provides **Services all around high quality PHP**

Outline

The problem

History

Requirements

Design

Summary

The problem

- Users installing software
  - Shared hosts
  - Dedicated servers
  - (Administrators)

The problem

- Users installing software
  - Shared hosts
  - Dedicated servers
  - (Administrators)
- The use case
  - Download installer package
  - Copy it to server
  - Configure installation
  - Install software

The problem

- ▶ Users installing software
  - ▶ Shared hosts
  - ▶ Dedicated servers
  - ▶ (Administrators)
- ▶ The use case
  - ▶ Download installer package
  - ▶ Copy it to server
  - ▶ Configure installation
  - ▶ Install software
- ▶ Nothing like this exists

Outline

History

- ▶ Christian Weiske blogged about the problem and his requirements
    - ▶ http://cweiske.de/tagebuch/ GenericPHPapplicationinstallers.htm

History

- ▶ Christian Weiske blogged about the problem and his requirements
  - ▶ http://cweiske.de/tagebuch/ GenericPHPapplicationinstallers.htm
- ▶ We need a proper installer for Arbit

History

- ▶ Christian Weiske blogged about the problem and his requirements
  - ▶ http://cweiske.de/tagebuch/ GenericPHPapplicationinstallers.htm
- ▶ We need a proper installer for Arbit
- ▶ We set up a wiki document for additional requirements and initial design
  - ▶ http://tracker.arbitracker.org/arbit/development_ wiki/view/Installer

History

- ▶ Christian Weiske blogged about the problem and his requirements
  - ▶ http://cweiske.de/tagebuch/ GenericPHPapplicationinstallers.htm
- ▶ We need a proper installer for Arbit
- ▶ We set up a wiki document for additional requirements and initial design
  - ▶ http://tracker.arbitracker.org/arbit/development_ wiki/view/Installer
- ▶ Discussed the problems at various (un-)conferences

History

- ▶ Christian Weiske blogged about the problem and his requirements
  - ▶ `http://cweiske.de/tagebuch/`
    `GenericPHPapplicationinstallers.htm`
- ▶ We need a proper installer for Arbit
- ▶ We set up a wiki document for additional requirements and initial design
  - ▶ `http://tracker.arbitracker.org/arbit/development_`
    `wiki/view/Installer`
- ▶ Discussed the problems at various (un-)conferences
- ▶ Merged feedback from phpBB, Typo3 and other software communities

Outline

General

- Handle "any" setup gracefully
  - PHP versions
  - Available extensions
  - PHP configurations

General

- ▶ Handle "any" setup gracefully
  - ▶ PHP versions
  - ▶ Available extensions
  - ▶ PHP configurations
- ▶ Ability to upgrade existing installations

General

- Handle "any" setup gracefully
  - PHP versions
  - Available extensions
  - PHP configurations
- Ability to upgrade existing installations
- Handling of task-graphs

Security

- Limit installation to one user
  - Require file with password created next to the installer
  - Limit installer to a single IP

Security

- ► Limit installation to one user
  - ► Require file with password created next to the installer
  - ► Limit installer to a single IP
- ► Drop rights after installation

Steps

1. Check system requirements
   - Extensions
   - Libaries
   - PHP configuration settings
   - Web Server setup

Steps

1. Check system requirements
   - ► Extensions
   - ► Libaries
   - ► PHP configuration settings
   - ► Web Server setup

2. Configure application
   - ► Accept license
   - ► Database
   - ► Application modules

Steps

1. Check system requirements
   - ▶ Extensions
   - ▶ Libaries
   - ▶ PHP configuration settings
   - ▶ Web Server setup

2. Configure application
   - ▶ Accept license
   - ▶ Database
   - ▶ Application modules

3. **Install application**
   - ▶ Copy data
   - ▶ Configure application
   - ▶ Set permissions

Outline

Design

- ▶ Installer consists of any number of tasks, each has
    - ▶ Requirements
    - ▶ Configuration options
    - ▶ Actions
    - ▶ (Verifications)

Design

- ▶ Installer consists of any number of tasks, each has
  - ▶ Requirements
  - ▶ Configuration options
  - ▶ Actions
  - ▶ (Verifications)
- ▶ Installer runs the start task
  - ▶ Each task may trigger any number of "dependant" tasks

Design

- ▶ Installer consists of any number of tasks, each has
    - ▶ Requirements
    - ▶ Configuration options
    - ▶ Actions
    - ▶ (Verifications)
- ▶ Installer runs the start task
    - ▶ Each task may trigger any number of "dependant" tasks
    - ▶ ... optionally based on configuration settings

Demo

Arbit Installer

Disecting the installer

```php
#!/usr/bin/php
<?php

require 'src/environment.php';

$installer = new aiInstaller(
    'Arbit',
    '0.3-alpha',
    'Arbit aims to provide a decent modern extensible multi project tracking
        tool. Features start with issue tracking and wiki and do not stop
        before code analysis and translation management.'
);
```

Disecting the installer

```
1   #!/usr/bin/php
2   <?php
3
4   require 'src/environment.php';
5
6   $installer = new aiInstaller(
7       'Arbit',
8       '0.3-alpha',
9       'Arbit aims to provide a decent modern extensible multi project tracking
                tool. Features start with issue tracking and wiki and do not stop
                before code analysis and translation management.'
10  );
```

Disecting the installer

```php
#!/usr/bin/php
<?php

require 'src/environment.php';

$installer = new aiInstaller(
    'Arbit',
    '0.3-alpha',
    'Arbit aims to provide a decent modern extensible multi project tracking
        tool. Features start with issue tracking and wiki and do not stop
        before code analysis and translation management.'
);
```

Adding a first task

```
12    // Configure legal task
13    $installer['legal'] = $legal = new aiTask(
14        'License',
15        'Please read and accept the license.'
16    );
17
18    $legal->configurations['license'] = new aiConfigurationLicense(
19        'Accept License',
20        'Please read and accept the GPL 3 license, under which Arbit is licensed.',
21        file_get_contents( __DIR__ . '/LICENSE' )
22    );
23    $legal->configurations['deps'] = new aiConfigurationDependency(
24        'Dependencies',
25        'Task dependencies.',
26        array( 'main' )
27    );
```

Adding a first task

```php
12    // Configure legal task
13    $installer['legal'] = $legal = new aiTask(
14        'License',
15        'Please read and accept the license.'
16    );
17
18    $legal->configurations['license'] = new aiConfigurationLicense(
19        'Accept License',
20        'Please read and accept the GPL-3 license, under which Arbit is licensed.',
21        file_get_contents( __DIR__ . '/LICENSE' )
22    );
23    $legal->configurations['deps'] = new aiConfigurationDependency(
24        'Dependencies',
25        'Task dependencies.',
26        array( 'main' )
27    );
```

Adding a first task

```
12    // Configure legal task
13    $installer['legal'] = $legal = new aiTask(
14        'License',
15        'Please read and accept the license.'
16    );
17
18    $legal->configurations['license'] = new aiConfigurationLicense(
19        'Accept License',
20        'Please read and accept the GPL 3 license, under which Arbit is licensed.',
21        file_get_contents( __DIR__ . '/LICENSE' )
22    );
23    $legal->configurations['deps'] = new aiConfigurationDependency(
24        'Dependencies',
25        'Task dependencies.',
26        array( 'main' )
27    );
```

Adding a first task

```php
12   // Configure legal task
13   $installer['legal'] = $legal = new aiTask(
14       'License',
15       'Please read and accept the license.'
16   );
17
18   $legal->configurations['license'] = new aiConfigurationLicense(
19       'Accept License',
20       'Please read and accept the GPL 3 license, under which Arbit is licensed.',
21       file_get_contents( __DIR__ . '/LICENSE' )
22   );
23   $legal->configurations['deps'] = new aiConfigurationDependency(
24       'Dependencies',
25       'Task dependencies.',
26       array( 'main' )
27   );
```

qafoo
passion for software quality

Adding a first task

```php
// Configure legal task
$installer['legal'] = $legal = new aiTask(
    'License',
    'Please read and accept the license.'
);

$legal->configurations['license'] = new aiConfigurationLicense(
    'Accept License',
    'Please read and accept the GPL 3 license, under which Arbit is licensed.',
    file_get_contents( __DIR__ . '/LICENSE' )
);
$legal->configurations['deps'] = new aiConfigurationDependency(
    'Dependencies',
    'Task dependencies.',
    array( 'main' )
);
```

Adding main task with requirements

```
29    // Configure main task
30    $installer['main'] = $main = new aiTask(
31        'Main_configuration',
32        'Main_configuration_for_arbit.'
33    );
34
35    $main->requirements = new aiRequirementsAnd(
36        new aiRequirementsAnd(
37            new aiRequirementPhpVersion( '5.3' ),
38            new aiRequirementPhpIniSetting( 'short_open_tags', false ),
39            new aiRequirementPhpIniSetting( 'memory_limit', '128M' ),
40            new aiRequirementPhpIniSetting( 'mbstring.func_overload', false ),
41            new aiRequirementPhpIniDateTimezone()
42        ),
43        new aiRequirementsAnd(
44            new aiRequirementPhpExtension( 'does_not_exist' ),
45            new aiRequirementPhpExtension( 'pcre' ),
46            new aiRequirementPhpExtension( 'spl' ),
47            new aiRequirementPhpExtension( 'date' ),
48            new aiRequirementPhpExtension( 'dom' ),
49            new aiRequirementPhpExtension( 'iconv' ),
50            new aiRequirementPhpExtension( 'json' ),
51            new aiRequirementPhpExtension( 'libxml' ),
52            new aiRequirementPhpExtension( 'session' )
53        ),
54        new aiRequirementSpace( '50M' )
55    );
```

Adding main task with requirements

```
29    // Configure main task
30    $installer['main'] = $main = new aiTask(
31        'Main_configuration',
32        'Main_configuration_for_arbit.'
33    );
34
35    $main->requirements = new aiRequirementsAnd(
36        new aiRequirementsAnd(
37            new aiRequirementPhpVersion( '5.3' ),
38            new aiRequirementPhpIniSetting( 'short_open_tags', false ),
39            new aiRequirementPhpIniSetting( 'memory_limit', '128M' ),
40            new aiRequirementPhpIniSetting( 'mbstring.func_overload', false ),
41            new aiRequirementPhpIniDateTimezone()
42        ),
43        new aiRequirementsAnd(
44            new aiRequirementPhpExtension( 'does_not_exist' ),
45            new aiRequirementPhpExtension( 'pcre' ),
46            new aiRequirementPhpExtension( 'spl' ),
47            new aiRequirementPhpExtension( 'date' ),
48            new aiRequirementPhpExtension( 'dom' ),
49            new aiRequirementPhpExtension( 'iconv' ),
50            new aiRequirementPhpExtension( 'json' ),
51            new aiRequirementPhpExtension( 'libxml' ),
52            new aiRequirementPhpExtension( 'session' )
53        ),
54        new aiRequirementSpace( '50M' )
55    );
```

Main task configurations

```php
$main->configurations['database'] = new aiConfigurationDatabase(
    'Database connection',
    'Database which stores the arbit data.',
    array(
        'couchdb' => 'Apache CouchDB document database',
        'mysql'   => 'MySQL database using the mysql driver',
    )
);

$main->configurations['email'] = new aiConfigurationString(
    'Admin E-Mail',
    'Mail address of administrator for error reporting.',
    'errors@example.org'
);
$main->configurations['sender'] = new aiConfigurationString(
    'Sender E-Mail',
    'Mail address used as the sender address for all mails sent out.',
    'Arbit <no-reply@example.org>'
);
$main->configurations['project'] = new aiConfigurationOptionalDependency(
    'Configure Project',
    'Configure a project directly inside the installer. Otherwse this can be
        done later by modifying the configuration files.',
    'project'
);
```

Main task actions

```
82   $main->actions [] = new aiActionPhync( array(
83       array(
84           new phyncMatcherTargetExists(),
85           array(
86               new phyncChmodAction( 0770 ),
87           ),
88       ),
89       array(
90           new phyncMatcherOr( array(
91               new phyncMatcherNot(
92                   new phyncMatcherTargetExists()
93               ),
94               new phyncMatcherSourceNewer(),
95           ) ),
96           array(
97               new phyncCopyAction(),
98               new phyncChmodAction( 0770 ),
99           ),
100      ),
101  ) );
```

Interlude: Phync

▶ Implements user-land PHP stream wrappers for different protocols
  ▶ FTP (using the PHP extension or sockets)
  ▶ Virtual File System (Memory)
  ▶ (SCP)

Interlude: Phync

- ▶ Implements user-land PHP stream wrappers for different protocols
    - ▶ FTP (using the PHP extension or sockets)
    - ▶ Virtual File System (Memory)
    - ▶ (SCP)
- ▶ Implements File system synchronization
    - ▶ Based on a set of rules, consisting of matchers and actions

Interlude: Phync

- ▶ Implements user-land PHP stream wrappers for different protocols
  - ▶ FTP (using the PHP extension or sockets)
  - ▶ Virtual File System (Memory)
  - ▶ (SCP)
- ▶ Implements File system synchronization
  - ▶ Based on a set of rules, consisting of matchers and actions
    - ▶ Matcher: Decides if the rule should be applied
    - ▶ Action: Executed if the rules applies

Interlude: Phync

- ▶ Implements user-land PHP stream wrappers for different protocols
  - ▶ FTP (using the PHP extension or sockets)
  - ▶ Virtual File System (Memory)
  - ▶ (SCP)
- ▶ Implements File system synchronization
  - ▶ Based on a set of rules, consisting of matchers and actions
    - ▶ Matcher: Decides if the rule should be applied
    - ▶ Action: Executed if the rules applies

Main task actions

```php
<?php

stream_wrapper_register( 'phyncftp', 'phyncFtpHandler' );
stream_wrapper_register( 'vfs', 'phyncVirtualFileSystemHandler' );

$syncer = new phyncSyncer();

$syncer->addRule(
    new phyncMatcherNot(
        new phyncMatcherTargetExists()
    ),
    array(
        new phyncCopyAction(),
        new phyncChmodAction( 0777 ),
    )
);
$syncer->addRule(
    new phyncMatcherFile(),
    array(
        new phyncChmodAction( 0666 ),
    )
);

$syncer->syncronize(
    'phyncftp://user@pass:localhost/subdir',
    'vfs://test/'
);
```

Main task actions

```php
<?php

stream_wrapper_register( 'phyncftp', 'phyncFtpHandler' );
stream_wrapper_register( 'vfs', 'phyncVirtualFileSystemHandler' );

$syncer = new phyncSyncer();

$syncer->addRule(
    new phyncMatcherNot(
        new phyncMatcherTargetExists()
    ),
    array(
        new phyncCopyAction(),
        new phyncChmodAction( 0777 ),
    )
);
$syncer->addRule(
    new phyncMatcherFile(),
    array(
        new phyncChmodAction( 0666 ),
    )
);

$syncer->syncronize(
    'phyncftp://user@pass:localhost/subdir',
    'vfs://test/'
);
```

Main task actions

```php
<?php

stream_wrapper_register( 'phyncftp', 'phyncFtpHandler' );
stream_wrapper_register( 'vfs', 'phyncVirtualFileSystemHandler' );

$syncer = new phyncSyncer();

$syncer->addRule(
    new phyncMatcherNot(
        new phyncMatcherTargetExists()
    ),
    array(
        new phyncCopyAction(),
        new phyncChmodAction( 0777 ),
    )
);
$syncer->addRule(
    new phyncMatcherFile(),
    array(
        new phyncChmodAction( 0666 ),
    )
);

$syncer->syncronize(
    'phyncftp://user@pass:localhost/subdir',
    'vfs://test/'
);
```

Main task actions

```php
<?php

stream_wrapper_register( 'phyncftp', 'phyncFtpHandler' );
stream_wrapper_register( 'vfs', 'phyncVirtualFileSystemHandler' );

$syncer = new phyncSyncer();

$syncer->addRule(
    new phyncMatcherNot(
        new phyncMatcherTargetExists()
    ),
    array(
        new phyncCopyAction(),
        new phyncChmodAction( 0777 ),
    )
);
$syncer->addRule(
    new phyncMatcherFile(),
    array(
        new phyncChmodAction( 0666 ),
    )
);

$syncer->syncronize(
    'phyncftp://user@pass:localhost/subdir',
    'vfs://test/'
);
```

Main task actions

```php
<?php

stream_wrapper_register( 'phyncftp', 'phyncFtpHandler' );
stream_wrapper_register( 'vfs', 'phyncVirtualFileSystemHandler' );

$syncer = new phyncSyncer();

$syncer->addRule(
    new phyncMatcherNot(
        new phyncMatcherTargetExists()
    ),
    array(
        new phyncCopyAction(),
        new phyncChmodAction( 0777 ),
    )
);
$syncer->addRule(
    new phyncMatcherFile(),
    array(
        new phyncChmodAction( 0666 ),
    )
);

$syncer->syncronize(
    'phyncftp://user@pass:localhost/subdir',
    'vfs://test/'
);
```

Main task actions

```
82   $main->actions[] = new aiActionPhync( array(
83       array(
84           new phyncMatcherTargetExists(),
85           array(
86               new phyncChmodAction( 0770 ),
87           ),
88       ),
89       array(
90           new phyncMatcherOr( array(
91               new phyncMatcherNot(
92                   new phyncMatcherTargetExists()
93               ),
94               new phyncMatcherSourceNewer(),
95           ) ),
96           array(
97               new phyncCopyAction(),
98               new phyncChmodAction( 0770 ),
99           ),
100      ),
101  ) );
```

Main task actions

```
102    $main−>actions [] = new aiActionIniConfig (
103        'config/main.ini',
104        array(
105            'email'           => 'main.email',
106            'type'            => 'main.database.extension',
107            'connection'      => 'main.database.url',
108            'options[from]'   => 'main.sender',
109        )
110    );
111    $main−>actions [] = new aiActionMkdir ( 'var/log', 0770 );
112    $main−>actions [] = new aiActionMkdir ( 'tmp', 0770 );
```

Main task actions

```
114    $main->postActions [] = new aiActionPhync( array(
115        array(
116            new phyncMatcherAnd( array(
117                new phyncMatcherDir(),
118                new phyncMatcherNot(
119                    new phyncMatcherPcre( '(/(?:var|tmp|log)/)' )
120                ),
121            ) ),
122            array(
123                new phyncChmodAction( 0550 ),
124            ),
125        ),
126        array(
127            new phyncMatcherFile(),
128            array(
129                new phyncChmodAction( 0440 ),
130            ),
131        ),
132    ) );
```

Packaging

```
212    $packager = new aiPackagerPhar();
213    $packager->package(
214        $installer,
215        new RecursiveRegexIterator(
216            new RecursiveRegexIterator(
217                new RecursiveRegexIterator(
218                    new RecursiveDirectoryIterator(
219                        '/home/kore/devel/arbit/trunk/src',
220                        FilesystemIterator::KEY_AS_PATHNAME | FilesystemIterator::
                            CURRENT_AS_FILEINFO
221                    ),
222                    aiPackager::createRejectRegExp( '/.svn/' ),
223                    RegexIterator::MATCH,
224                    RegexIterator::USE_KEY
225                ),
226                aiPackager::createRejectRegExp( '/var/' ),
227                RegexIterator::MATCH,
228                RegexIterator::USE_KEY
229            ),
230            aiPackager::createRejectRegExp( '/tmp/' ),
231            RegexIterator::MATCH,
232            RegexIterator::USE_KEY
233        ),
234        'arbit-0.3-alpha-installer.phar'
235    );
```

Outline

Summary

- Extensible task based fully configurable installer
  - Build your own installer with it
  - Current state: svn:
    //arbitracker.org/arbit/projects/installer/trunk
    - Will probably move to github soonish:
      svn://github.org/arbitracker

Summary

- Extensible task based fully configurable installer
  - Build your own installer with it
  - Current state: svn:
    //arbitracker.org/arbit/projects/installer/trunk
    - Will probably move to github soonish:
      svn://github.org/arbitracker
- TODO
  - Security features

Summary

- ► Extensible task based fully configurable installer
  - ► Build your own installer with it
  - ► Current state: svn:
    //arbitracker.org/arbit/projects/installer/trunk
    - ► Will probably move to github soonish:
      svn://github.org/arbitracker
- ► TODO
  - ► Security features
  - ► Command line user interface
    - Easier to use it as an "updater"
    - A non-PHAR packager
    - and replay configuration
    - like configuration through XML

Summary

- ▶ Extensible task based fully configurable installer
  - ▶ Build your own installer with it
  - ▶ Current state: svn:
    //arbitracker.org/arbit/projects/installer/trunk
    - ▶ Will probably move to github soonish:
      svn://github.org/arbitracker
- ▶ TODO
  - ▶ Security features
  - ▶ Command line user interface
  - ▶ Make it easier to use it as an "updater"

Summary

- ▶ Extensible task based fully configurable installer
    - ▶ Build your own installer with it
    - ▶ Current state: svn:
      //arbitracker.org/arbit/projects/installer/trunk
        - ▶ Will probably move to github soonish:
          svn://github.org/arbitracker
- ▶ TODO
    - ▶ Security features
    - ▶ Command line user interface
    - ▶ Make it easier to use it as an "updater"
    - ▶ Provide non-PHAR packager

Summary

- Extensible task based fully configurable installer
  - Build your own installer with it
  - Current state: svn:
    //arbitracker.org/arbit/projects/installer/trunk
    - Will probably move to github soonish:
      svn://github.org/arbitracker
- TODO
  - Security features
  - Command line user interface
  - Make it easier to use it as an "updater"
  - Provide non-PHAR packager
  - Store and replay configuration

Summary

- Extensible task based fully configurable installer
  - Build your own installer with it
  - Current state: svn:
    //arbitracker.org/arbit/projects/installer/trunk
    - Will probably move to github soonish:
      svn://github.org/arbitracker
- TODO
  - Security features
  - Command line user interface
  - Make it easier to use it as an "updater"
  - Provide non-PHAR packager
  - Store and replay configuration
  - Installer configuration through XML

Thanks for listening

Please rate this talk at
http://joind.in/2986
and / or give me some feedback right now!

Thanks for listening

Please rate this talk at
`http://joind.in/2986`
and / or give me some feedback right now!

Stay in touch

► Kore Nordmann / <kore@qafoo.com> / @koredn / @qafoo
► **Rent a PHP quality expert**:
  `http://qafoo.com`