

# Advanced project tracking with Arbit

## BEPHPUG Sep. 2010

Daniel Fahlke aka Flyingmana

September 7, 2010

## About me

- ▶ Daniel Fahlke aka Flyingmana  
([<flyingmana@googlemail.com>](mailto:flyingmana@googlemail.com),  
[<flyingmana@arbitracker.org>](mailto:flyingmana@arbitracker.org))
- ▶ erster Kontakt mit PHP in 2005
- ▶ aktiv eingetaucht in die PHP Szene seit 2009
- ▶ bisherige Projekte: nur Arbit

## Was ist Arbit?

- ▶ Ein Projekt Tracker

## Was ist Arbit?

- ▶ Ein Projekt Tracker
  - ▶ Issue Tracker

## Was ist Arbit?

- ▶ Ein Projekt Tracker
  - ▶ Issue Tracker
  - ▶ Source Code Viewer

## Was ist Arbit?

- ▶ Ein Projekt Tracker
  - ▶ Issue Tracker
  - ▶ Source Code Viewer
  - ▶ PHP Werkzeuge

## Was ist Arbit?

- ▶ Ein Projekt Tracker
  - ▶ Issue Tracker
  - ▶ Source Code Viewer
  - ▶ PHP Werkzeuge
    - ▶ PHPUnit

## Was ist Arbit?

- ▶ Ein Projekt Tracker
  - ▶ Issue Tracker
  - ▶ Source Code Viewer
  - ▶ PHP Werkzeuge
    - ▶ PHPUnit
    - ▶ PHPloc
    - ▶ PHPdoc
    - ▶ PHP Code Sniffer
  - ▶ zusammengefasst, alles für eine Kontinuierliche Integration

# CI in Arbit



Wieso nichts vorhandenes nehmen?

## Wieso nichts vorhandenes nehmen?

- ▶ Was gibt es bereits?

## Wieso nichts vorhandenes nehmen?

- ▶ Was gibt es bereits?
  - ▶ Hudson
  - ▶ CruiseControl(`phpUnderControl`)

## Wieso nichts vorhandenes nehmen?

- ▶ Was gibt es bereits?
  - ▶ Hudson
  - ▶ CruiseControl(phpUnderControl)
    - ▶ beides Java

## Ziele von Arbit

## Ziele von Arbit

- ▶ komplett in PHP

## Ziele von Arbit

- ▶ komplett in PHP
- ▶ hohe Qualität/Sauberkeit des Code selbst

## Ziele von Arbit

- ▶ komplett in PHP
- ▶ hohe Qualität/Sauberkeit des Code selbst
- ▶ hoher Grad an Modularität

## Ziele von Arbit

- ▶ komplett in PHP
- ▶ hohe Qualität/Sauberkeit des Code selbst
- ▶ hoher Grad an Modularität
- ▶ Unterstützung mehrerer Projekte

## Dashboard



### Arbit - project tracking

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.

- o issue\_tracker: 62 open issues of 163 issues in issue tracker.
- o browse\_source: Checked out revision #1705 of project source.
- o documentation: 6 pages available in wiki.
- o phpunit: 630 tests run with PHPUnit, 0 failures (163.67 seconds, 864 assertions).
- o development\_wiki: 11 pages available in wiki.

PQI\*



### PHPillow - PHP CouchDB connector

PHPillow is a PHP CouchDB wrapper, which lays on top of the Couch and offers even more comfort laying anything into the database.

- o issue\_tracker: 5 open issues of 9 issues in issue tracker.
- o browse\_source: Checked out revision #152 of project source.
- o phpunit: 165 tests run with PHPUnit, 2 failures (3.44 seconds, 238 assertions).

PQI\*



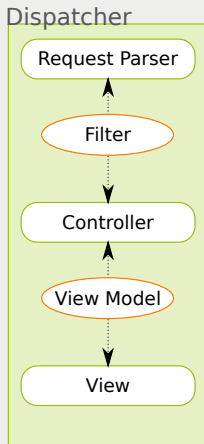
## Ein paar Details zum Projekt

- ▶ gestartet im Februar 2008
- ▶ noch im Alpha Status
- ▶ momentane Version 0.3
- ▶ bereits "benutzbar"
- ▶ professioneller Support durch Qafoo
- ▶ Momentane Abhängigkeiten
  - ▶ PHP  $\geq$  5.3
  - ▶ CouchDB  $\geq$  0.9
- ▶ Nutzt zu einem großen Teil die Zeta Components, ein wenig Symphony und ein wenig Zend

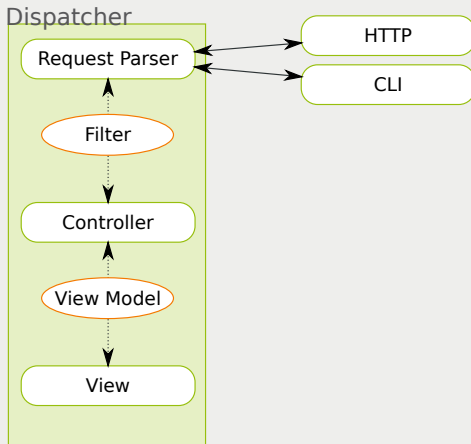
## Subprojekte

- ▶ PHPillow
  - ▶ Objekt Orientierter CouchDB Wrapper
- ▶ VCSWrapper
  - ▶ Abstraktions Ebene für Versions Kontroll Systeme
  - ▶ Unterstützt bisher:
    - ▶ SVN über das cli Interface oder PHP extension
    - ▶ CVS über das cli Interface
    - ▶ GIT auch über das cli Interface
    - ▶ Mercurial über das hg-cli Interface
    - ▶ Archive über die ZipArchive Klasse in PHP
- ▶ Periodic
  - ▶ “Periodic is a fully unit tested PHP based task runner”
- ▶ PHP Commit Hooks
- ▶ arbit-result-printer

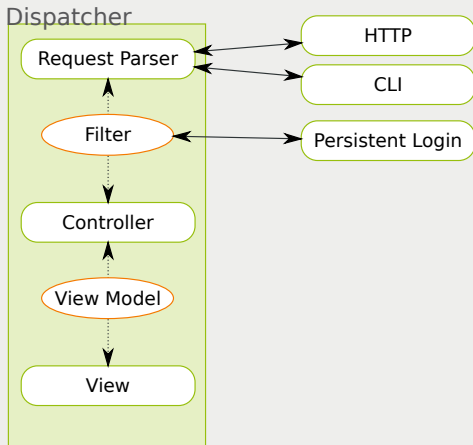
## Grundstruktur der Architektur



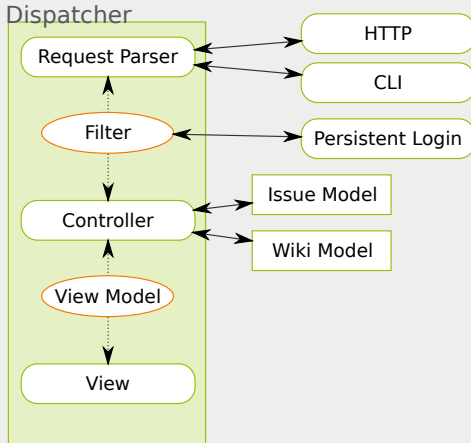
## Grundstruktur der Architektur



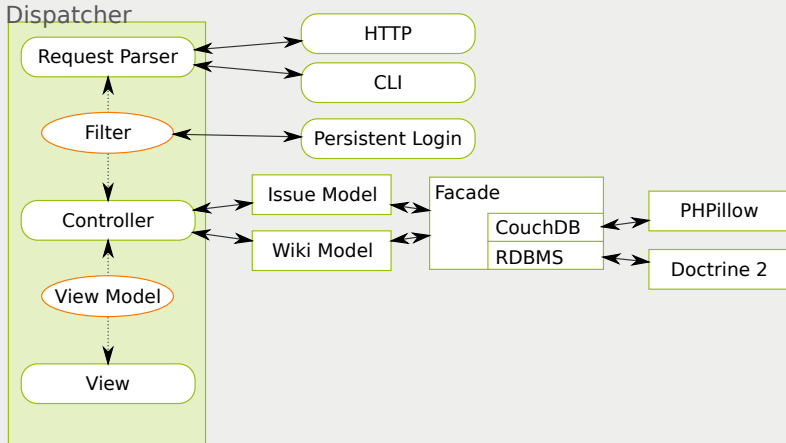
## Grundstruktur der Architektur



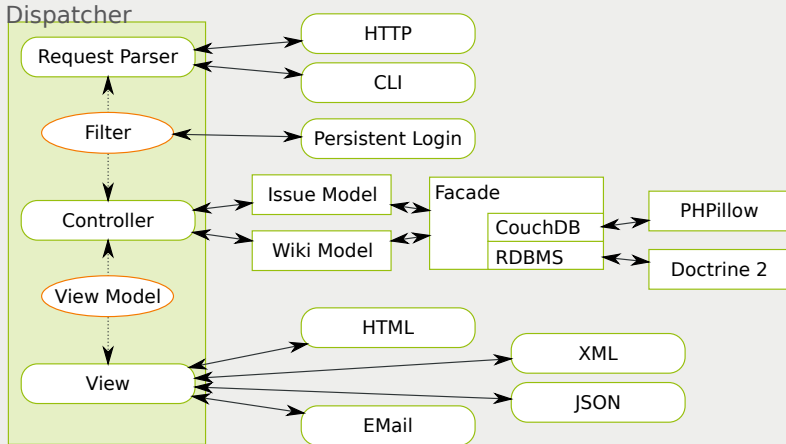
## Grundstruktur der Architektur



# Grundstruktur der Architektur



# Grundstruktur der Architektur



Welche Grundfunktionen bietet Arbit den Modulen?

## Welche Grundfunktionen bietet Arbit den Modulen?

- ▶ User Authentifikation (ist eigentlich auch ein Modul)

## Welche Grundfunktionen bietet Arbit den Modulen?

- ▶ User Authentifikation (ist eigentlich auch ein Modul)
- ▶ Kommunikation über Signal Slots

## Welche Grundfunktionen bietet Arbit den Modulen?

- ▶ User Authentifikation (ist eigentlich auch ein Modul)
- ▶ Kommunikation über Signal Slots
- ▶ Datenbankverbindung

## Welche Grundfunktionen bietet Arbit den Modulen?

- ▶ User Authentifikation (ist eigentlich auch ein Modul)
- ▶ Kommunikation über Signal Slots
- ▶ Datenbankverbindung
- ▶ Rechteverwaltung
- ▶ Zugriff auf Aktionen via CLI und periodic

## Wie ist ein Modul aufgebaut?

- ▶ 2 grundlegende Dateien

## Wie ist ein Modul aufgebaut?

- ▶ 2 grundlegende Dateien
  - ▶ autoload.php

## Wie ist ein Modul aufgebaut?

- ▶ 2 grundlegende Dateien
  - ▶ autoload.php
    - ▶ ein Array aller Modulklassen mit Pfad zur Datei

## Wie ist ein Modul aufgebaut?

- ▶ 2 grundlegende Dateien
  - ▶ autoload.php
    - ▶ ein Array aller Modulklassen mit Pfad zur Datei
  - ▶ definition.php

## Wie ist ein Modul aufgebaut?

- ▶ 2 grundlegende Dateien
  - ▶ autoload.php
    - ▶ ein Array aller Modulklassen mit Pfad zur Datei
  - ▶ definition.php
    - ▶ eine Klasse die arbitModuleDefintion erweitert

## definition.php

### arbitModuleSourceDefintion

```
<?php
```

```
protected $properties = array(
    'autoload'           => null,
    'permissions'       => array(
        'source_manage' => 'User_is_allowed_perform_source_management_task,_like_check',
        'source_view_source' => 'User_is_allowed_to_view_the_source_code_in_the_web_interf',
    ),
    'slots'             => array(
        'sourceAnnotate' => 'arbitModuleSourceController::annotateSource',
    ),
    'signals'          => array(
        'sourceAnnotate' => 'Annotate_a_source_file_[,line_[,position]]_with_addition',
        'sourceInitialized' => 'The_source_checkout_has_been_initialized.',
        'sourceUpdated' => 'The_source_checkout_has_been_updated.',
    ),
    'templateDirectory' => 'templates/',
    'controller'        => 'arbitModuleSourceController',
    'path'              => __DIR__,
);
?>
```

## definition.php

### arbitModuleSourceDefintion

```

<?php

protected $caches = array(
    'source_vcs' => array(
        'path' => 'source/model/',
        'ttl' => arbitCache::INFINITE,
    ),
);
protected $facades = array(
    'couchdb' => array(
        'source_annotation' => 'arbitCouchDbSourceAnnotationFacade',
    ),
);
protected $couchDbDocuments = array(
    'source_annotation' => 'arbitBackendCouchDbSourceAnnotationDocument',
);
protected $couchDbViews = array(
    'source_annotations' => 'arbitBackendCouchDbSourceAnnotationView',
);

?>

```

## definition.php

### arbitModuleSourceDefintion

```
<?php
```

```
protected $viewModels = array(
    'arbitViewXHtmlHandler' => array(
        'arbitSourceCheckoutViewModel' => 'arbitModuleSourceXHtmlHandler::showCheckoutModel',
        'arbitSourceDirectoryViewModel' => 'arbitModuleSourceXHtmlHandler::showDirectoryModel',
        'arbitSourceFileViewModel' => 'arbitModuleSourceXHtmlHandler::showFileModel',
        'arbitSourceHistoryModel' => 'arbitModuleSourceXHtmlHandler::showHistoryModel',
        'arbitSourceDiffViewModel' => 'arbitModuleSourceXHtmlHandler::showDiffModel',
        'arbitSourceAnnotationsViewModel' => 'arbitModuleSourceXHtmlHandler::showAnnotationsModel',
    ),
    'arbitViewTextHandler' => array(
        'arbitSourceCheckoutViewModel' => 'arbitModuleSourceTextHandler::showCheckoutModel',
    ),
);

protected $commands = array(
    'source.update' => 'arbitModuleSourceUpdateCommand',
    'source.prime-cache' => 'arbitModuleSourcePrimeCacheCommand',
    'source.clean-metadata-cache' => 'arbitModuleSourceCleanMetadataCacheCommand',
);
?>
```