Arbit - next generation project tracking

http://joind.in/1446

Kore Nordmann
<kore@php.net>
@koredn

May 14, 2010

About me

- Kore Nordmann, <kore@php.net>
- Long time PHP developer
- Studies computer science in Dortmund, currently writing thesis
- Currently founding *Qafoo* (contact@qafoo.com / @qafoo)
- Active open source developer:
  - eZ Components (Graph, WebDav, Document), *Arbit*, PHPUnit, Torii, *PHPillow*, KaForkL, Image 3D, WCV, ...

Outline

Introduction

Motivation

Current state

Architectural gems

Sub-Projects

The future

QA

What is arbit?

- An issue tracker on steroids.

What is arbit?

- An issue tracker on steroids.
    - A “project tracker”

Outline

Introduction

Motivation

Current state

Architectural gems

Sub-Projects

The future

QA

Motivation

- Yet another issue tracker - why?

Motivation

- Yet another issue tracker - why?
- There are already so many out there. . .
  - Trac
  - Redmine
  - Mantis
  - Jira
  - Bugzilla
  - Flyspray
  - . . .

Motivation

- Yet another issue tracker - why?
- There are already so many out there...
  - Trac
  - Redmine
  - Mantis
  - Jira
  - Bugzilla
  - Flyspray
  - ...
- So, why develop another one?

Motivation

- Yet another issue tracker - why?
- There are already so many out there. . .
  - Trac
  - Redmine
  - Mantis
  - Jira
  - Bugzilla
  - Flyspray
  - . . .
- So, why develop another one?
  - Anybody got an idea?
  - What do you hate most about them?

Initial roadmap

- Started in Februrary 2008
- Primary development goals

Initial roadmap

- Started in Februrary 2008
- Primary development goals
    - Clean extensible PHP code

Initial roadmap

- Started in Februrary 2008
- Primary development goals
  - Clean extensible PHP code
  - Native, easy multi-project support

Initial roadmap

- Started in Februrary 2008
- Primary development goals
  - Clean extensible PHP code
  - Native, easy multi-project support
  - Modularized, for custom extensions

Initial roadmap

- Started in Februrary 2008
- Primary development goals
  - Clean extensible PHP code
  - Native, easy multi-project support
  - Modularized, for custom extensions
- Originally planned feature set

Initial roadmap

- Started in Februrary 2008
- Primary development goals
  - Clean extensible PHP code
  - Native, easy multi-project support
  - Modularized, for custom extensions
- Originally planned feature set
  - Issue tracker, wiki, source browsing, notifications, . . .

Initial roadmap

- Started in Februrary 2008
- Primary development goals
  - Clean extensible PHP code
  - Native, easy multi-project support
  - Modularized, for custom extensions
- Originally planned feature set
  - Issue tracker, wiki, source browsing, notifications, . . .
  - Integrate Continuous Integration (CI) with issue tracking

Integrating CI

- What does CI mean?

Integrating CI

- What does CI mean?
  - Running tests on each commit / every hour

Integrating CI

- What does CI mean?
  - Running tests on each commit / every hour
  - Analyze source code for defects

Integrating CI

- What does CI mean?
  - Running tests on each commit / every hour
  - Analyze source code for defects
  - Build releases on build success

Integrating CI

- What does CI mean?
  - Running tests on each commit / every hour
  - Analyze source code for defects
  - Build releases on build success
  - Report failures

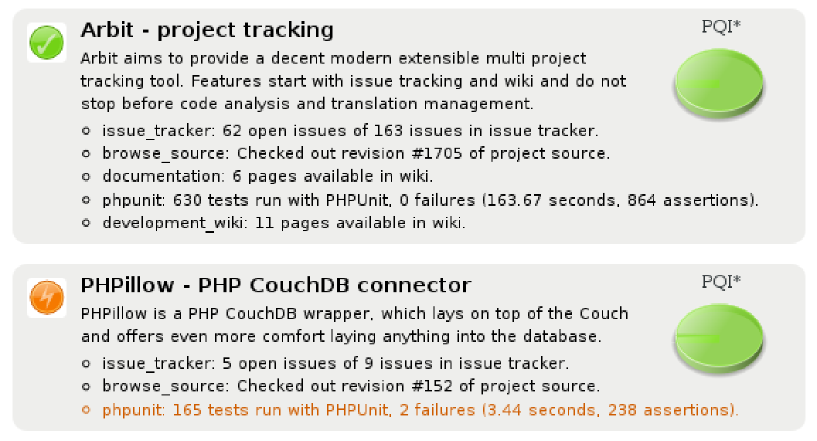
Integrating CI

- What does CI mean?
  - Running tests on each commit / every hour
  - Analyze source code for defects
  - Build releases on build success
  - Report failures
- We got all of this in Hudson, phpUnderControl, . . .

Integrating CI

- What does CI mean?
  - Running tests on each commit / every hour
  - Analyze source code for defects
  - Build releases on build success
  - Report failures
- We got all of this in Hudson, phpUnderControl, . . .
- Benefits of integration with issue trackers
  - Combine knowledge about bug statistics with source metrics
  - Know “everything” about a project at one glance

Dashboard



✓ Arbit - project tracking

PQI*

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.

- issue_tracker: 62 open issues of 163 issues in issue tracker.
- browse_source: Checked out revision #1705 of project source.
- documentation: 6 pages available in wiki.
- phpunit: 630 tests run with PHPUnit, 0 failures (163.67 seconds, 864 assertions).
- development_wiki: 11 pages available in wiki.

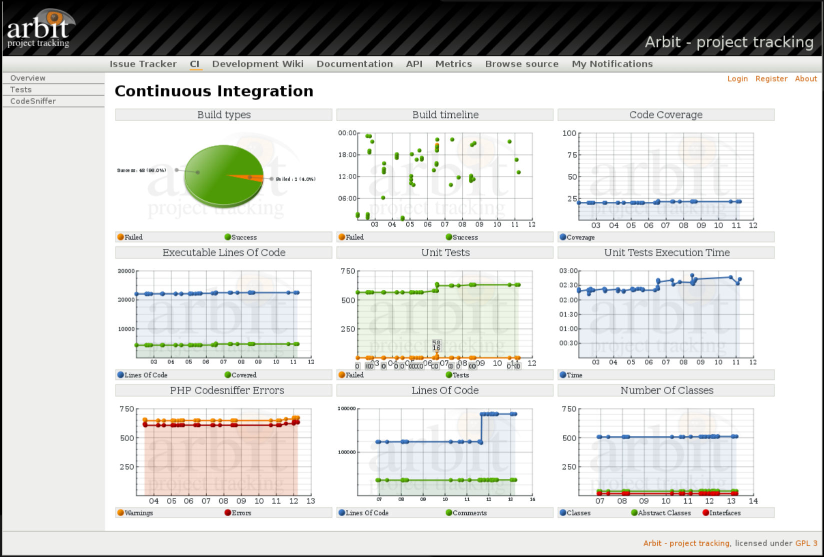
PHPillow - PHP CouchDB connector

PQI*

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

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

CI in Arbit



arbit
project tracking

Arbit - project tracking

Issue Tracker CI Development Wiki Documentation API Metrics Browse source My Notifications

Overview
Tests
CodeSniffer

Login Register About

Continuous Integration

Arbit - project tracking, licensed under GPL 3

Outline

Introduction

Motivation

Current state

Architectural gems

Sub-Projects

The future

QA

Dependencies

- PHP 5.3
- CouchDB 0.9

- Installation
  - Download PHAR or archive
  - Installation instructions: `http://tracker.arbitracker.org/arbit/documentation/view/InstallationGuide`

Dependencies

- PHP 5.3
- CouchDB 0.9
  - Alternative RDBMS backend (MySQL, PostgreSQL, ...) is being worked on
    - Will probably use Doctrine 2
- Installation
  - Download PHAR or archive
  - Installation instructions: `http://tracker.arbitracker.org/arbit/documentation/view/InstallationGuide`

Contribute

- Already contributing:
  - Kore Nordmann (kore)
  - Jordi Boggiano (seldaek)
  - Manuel Pichler (mapi)
  - Arne Nordmann (norro)
  - Jakob Westhoff (jakob)
  - Hans-Christian Otto (hco)
  - Tobias Schlitt (toby)
  - Tobias Tom (tobias)
- We would love to welcome you in the community around arbit: `http://arbitracker.org/arbit/participate.html`

Already working

- Issue tracker
- Wiki
- Source browser
- Source metrics rendering (phpDepend)
- API documentation rendering
- Continuous Integration (CI) (*experimental*)
    - PHPUnit
    - PHPCodeSniffer
    - phploc

Already working

- Issue tracker
- Wiki
- Source browser
- Source metrics rendering (phpDepend)
- API documentation rendering
- Continuous Integration (CI) (*experimental*)
    - PHPUnit
    - PHPCodeSniffer
    - phploc
- Demo!

Outline

Introduction

Motivation

Current state

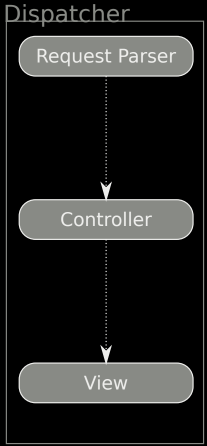
Architectural gems

Sub-Projects

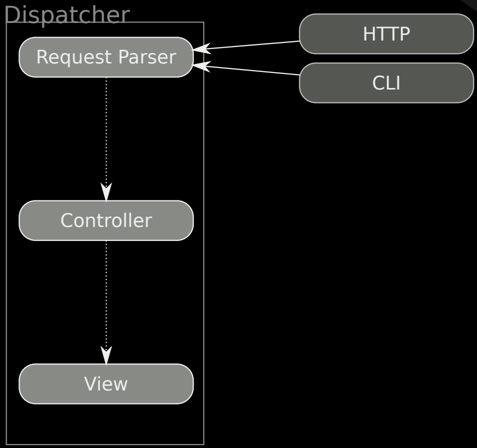
The future

QA

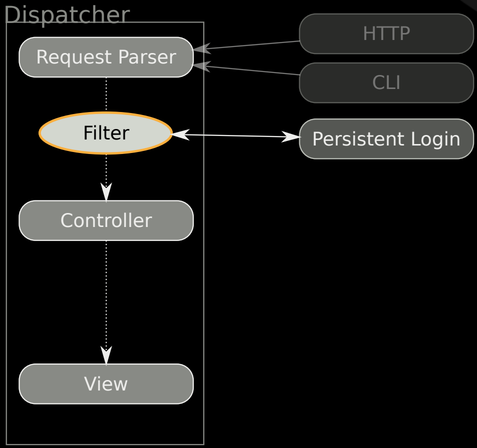
Arbit basic architecture



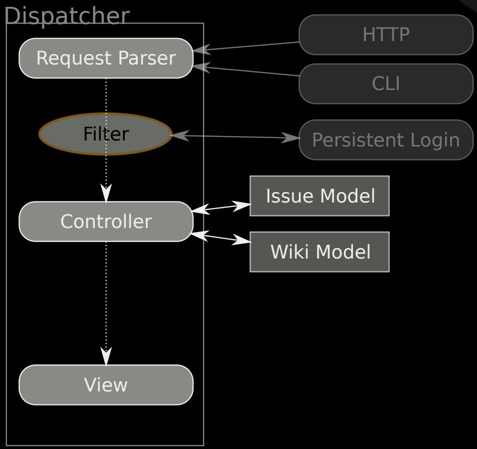
Arbit basic architecture



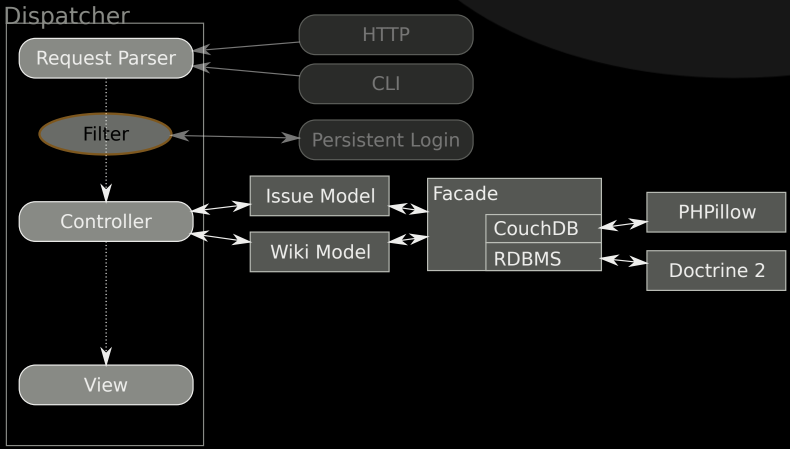
Arbit basic architecture



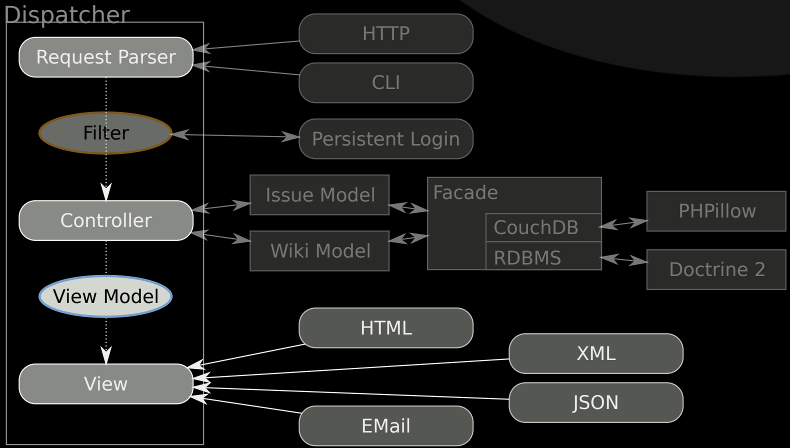
Arbit basic architecture



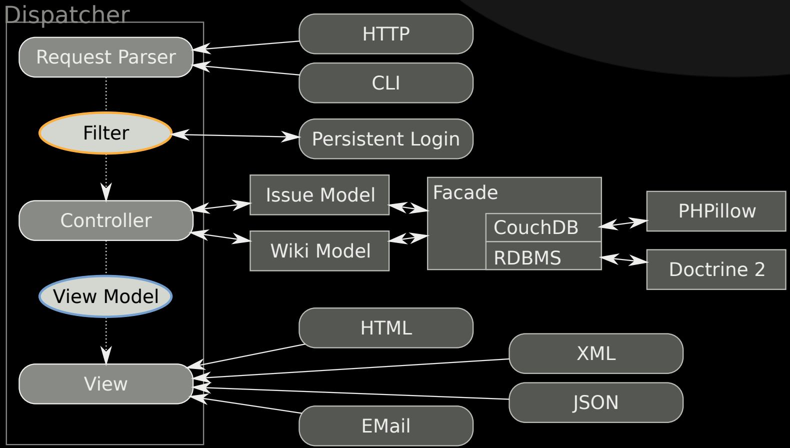
Arbit basic architecture



Arbit basic architecture



Arbit basic architecture



Module communication

- How can modules communicate?
  - No hard dependencies.
  - New modules should be able to interact with existing ones.

Module communication

- How can modules communicate?
  - No hard dependencies.
  - New modules should be able to interact with existing ones.
- Signal Slot

Signal slot

```php
<?php

$handler = new arbitSignalSlot();

$handler->register( 'signalA', 'myModule::handleSignal' );
$handler->register( 'signalA', 'yourModule::handleSignal' );

// In module c
$handler->emit( 'signalA', array( /* data */ ) );

// Now all modules registerd for this signal are called with the provided dataa

?>
```

Practical example

- Inform other modules about source code updates

```php
// modules/source/classes/controller.php +300
if ( $checkout->update( /* ... */ ) )
{
    // ...
    arbitSignalSlot::emit(
        'sourceUpdated',
        new arbitSourceUpdatedStruct( $checkout )
    );
}
```

Practical example

- Other modules can specify signals to listen for

```
// modules/phpunit/definition.php
class arbitModulePhpunitDefintion extends arbitModuleDefintion
{
    // ...
    protected $properties = array(
        // ...
        'slots'                 => array(
            'sourceUpdated' => 'arbitModulePhpunitController::sourceUpdated',
        ),
        // ...
    );
```

- Signals are sent for

Practical example

- Other modules can specify signals to listen for

```
// modules/phpunit/definition.php
class arbitModulePhpunitDefintion extends
        arbitModuleDefintion
{
        // ...
        protected $properties = array(
                // ...
                'slots'                   => array(
                        'sourceUpdated' => '
                                arbitModulePhpunitController::
                                sourceUpdated',
                ),
                // ...
        );
```

- All signals can also be converted into notifications (mail, jabber, ...)
- Signals are sent for

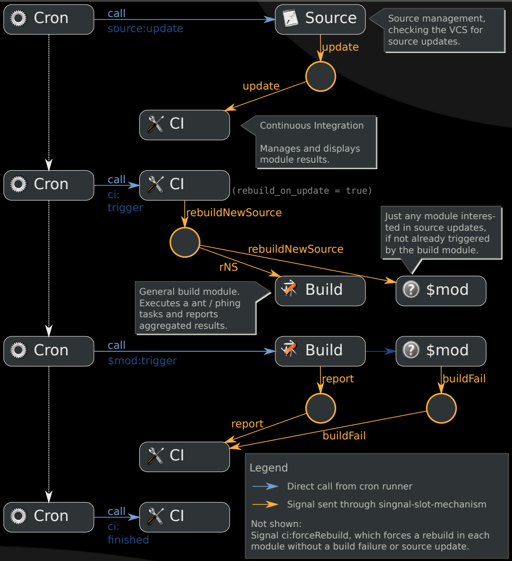
Practical example

- Other modules can specify signals to listen for

```
// modules/phpunit/definition.php
class arbitModulePhpunitDefintion extends
    arbitModuleDefintion
{
    // ...
    protected $properties = array(
        // ...
        'slots'           => array(
            'sourceUpdated' => '
                arbitModulePhpunitController::
                sourceUpdated',
        ),
        // ...
    );
```

- All signals can also be converted into notifications (mail, jabber, ...)
- Signals are sent for “everything”

Signal handling of the CI module



Source management, checking the VCS for source updates.

Continuous Integration

Manages and displays module results.

Just any module interested in source updates, if not already triggered by the build module.

General build module. Executes a ant / phing tasks and reports aggregated results.

Legend

→ Direct call from cron runner

→ Signal sent through singnal-slot-mechanism

Not shown:
Signal ci:forceRebuild, which forces a rebuild in each module without a build failure or source update.

Outline

Introduction

Motivation

Current state

Architectural gems

Sub-Projects

The future

QA

Periodic

- Task runner implemented in PHP
  - Can parse cron-tables (vixie-cron dialect)
  - Can handle parallel running cron tasks
- Design document: `http://arbitracker.org/periodic/design.html`
- Website: `http://arbitracker.org/periodic.html`

VCS-Wrapper

- Abstraction layer for version control system (VCS) read access
  - Implements support for: SVN, CVS, Git, Mercurial, (Archives)
  - Interfaces reflect which features are supported in which VCS
  - Supports: History, diffs, blaming, listings, ...
- Website: `http://arbitracker.org/vcs_wrapper.html`

PHPillow

- Lightweight PHP CouchDB library
- Website: `http://arbitracker.org/phpillow.html`

PHPillow

- Lightweight PHP CouchDB library
- Features
  - Simple document validation constraints
  - Automatic synchronization of views
  - Automatic versioning of documents
  - couchdb-python compatible tool for dump and import

- Website: `http://arbitracker.org/phpillow.html`

PHPillow

- Lightweight PHP CouchDB library
- Features
  - Simple document validation constraints
  - Automatic synchronization of views
  - Automatic versioning of documents
  - couchdb-python compatible tool for dump and import
- Different connection handlers
  - PHP HTTP stream wrapper
  - Custom HTTP protocol implementation
- Website: `http://arbitracker.org/phpillow.html`

Arbit installer

- Web installer for PHP projects
- Design phase just finished
  - Results at: `http://tracker.arbitracker.org/arbit/development_wiki/view/Installer`
  - Please check if it would work for *your* application
- Major feature of next arbit release

Outline

- Introduction
- Motivation
- Current state
- Architectural gems
- Sub-Projects
- The future
- QA

The future

- Next important tasks
  - Implement easy-to-use web installer
  - Refactor and stabilize CI related modules
  - Implement a RDBMS backend
- Current roadmap: `http://tracker.arbitracker.org/arbit/issue_tracker/roadmap`
- Qafoo will provide commercial support / consulting / training for arbit, once founded
  - Will still stay a fully cummunity-driven Open-Source project

Arbit feedback

- What would *you* like to see in arbit?
  - Special requirements
  - Important features

Outline

Introduction

Motivation

Current state

Architectural gems

Sub-Projects

The future

QA

Resources

- Website: `http://arbitracker.org/news.html`
- Issue tracker: `http://tracker.arbitracker.org/`
- CI: `http://tracker.arbitracker.org/arbit/ci`

The end

- Open questions?
- Further remarks?
- Contact
  - Mail: <kore@php.net>
  - Web: http://kore-nordmann.de/ (Slides will be available here soonish)
  - Twitter: http://twitter.com/koredn
  - Comment: http://joind.in/1446