FoolFuuka

Release 2.1.0

January 15, 2015

Introduction

FoolFuuka was designed to accomplish two different goals: (1) function as an image-based bulletin board software, and (2) provide a functional front-end interface for *Asagi*. In order to accomplish the latter, you'll need to prepare and configure your server environment properly.

1.1 Features

Requirements

2.1 Web Server

You'll need to install one of the following web servers to allow users to access the web interface:

- Apache
- nginx (Recommended)

2.2 PHP/HHVM

While FoolFuuka is designed to work on PHP 5.4 or newer, it is also compatible with HHVM.

2.3 Database

FoolFuuka supports the following database servers:

- MariaDB 5.5+ (Recommended)
- MySQL 5.5+

Note: When dealing with large amounts of data, we recommend using the `TokuDB` storage engine for all board tables which is available with MariaDB. This mainly applies to deployments used to provide a front-end for Asagi and archives.

2.4 Search Server

FoolFuuka requires an additional software to index and serve search results to users. This is achieved with Sphinx Open Source Search Server which must be installed to provide users with search capabilities.

Note: FoolFuuka includes a Sphinx configuration file generator in the web administration interface and should be used to generate a new sphinx configuration file accordingly.

2.5 Git/Composer

FoolFuuka utilizes both Git and Composer to manage all of its dependencies, software updates, and web assets.

Installation

3.1 Install

3.1.1 Via Composer

```
$ composer create-project foolz/foolfuuka foolfuuka
$ cd foolfuuka
$ composer dump-autoload --optimize
```

3.1.2 Via Source Code

```
$ git clone https://github.com/FoolCode/FoolFuuka foolfuuka
$ cd foolfuuka
$ git checkout 2-1-stable
$ composer install --optimize
```

3.2 Upgrading

```
$ git fetch --all
$ git checkout 2-1-stable
$ git pull
$ composer update --optimize
```

Note: The commands provided above will only upgrade the FoolFuuka code. You may be required to complete some additional steps to completely upgrade FoolFuuka to the next version. Please consult the upgrade guides to ensure that the upgrade process is done properly.

See also:

Upgrade Guide

Configuration

While most of the configurable settings are exposed through the web administrative panel, there are still a number of settings that can only be modified manually due to security concerns.

4.1 Administrative Panel

4.2 Config Files

In this documentation, we will use `dot` notations to specify the location of the config key being listed.

4.2.1 FoolFrame

These config files are used to manage many of the core settings in the FoolFuuka framework. You can locate these files in the following path after installation: `vendor/foolz/foolframe/config/`.

Note: We recommend that you copy these config files into and edit them in the following directory: `app/foolz/foolframe/config/`. The software is designed to load these files instead of the default config files.

cache

config

db

foolauth

4.2.2 FoolFuuka

These config files are used to manage many of the core settings in FoolFuuka that we considered very important and shouldn't be exposed in the web interface. You can locate these files in the following path after installation: `assets/config/`.

Note: We recommend that you copy these config files into and edit them in the following directory: `app/foolz/foolfuuka/config/`. The software is designed to load these files instead of the default config files.

config

comment.secure_tripcode_salt

Scope:	COMMENT
Type:	STRING

This is the salt used to for secure tripcodes. It is recommend that this salt key be changed when exposed or kept consistent between installations.

media.filecheck

Scope:	MEDIA
Type:	BOOLEAN
Default:	true

Checks if the media file exists on the disk. The setting does impact disk performance when enabled due to lstat calls for each file being checked.

Value	Effect
true	enables the check and returns a generated link based on file existence
false	disables the check and returns a full link

foolauth

User Guide

5.1 Asagi

5.1.1 Requirements

- Asagi
- JDK 7
- Maven

5.1.2 Compiling Asagi

```
$ git clone https://github.com/FoolCode/asagi.git
$ mvn package assembly:single
```

5.1.3 Configuring Asagi

Asagi uses a JSON configuration file named `asagi.json`. An example configuration file is included within thegit repository as `asagi.json.example`.

> **Warning:** If you are using MySQL/MariaDB as your database server, you must set the `character_set_server` setting under the `[mysqld]` section to `utf8mb4` in your `my.cnf` file. This will allow you to properly store multi-byte unicode characters properly.

Note: We recommend that you configure Asagi to use its own database and configure your database server to allow the account created for FoolFuuka to have full access to the Asagi database as well.

5.1.4 Running Asagi

```
$ java -Xmx256m -XX:+UseParNewGC -XX:MaxPermSize=24m -jar asagi.jar
```

Note: We strongly recommend the usage of `screen` or `tmux` with Asagi. Also, you may be required to adjust the `Xmx` and `XX:MaxPermSize` values accordingly.

5.1.5 Configuring FoolFuuka

> **Warning:** It is very crucial that you configure and run/restart Asagi before adding the board to FoolFuuka. This will allow Asagi to create the board tables properly with some additional steps that aren't included in FoolFuuka. If this is not done, the board tables will not be populated properly.

You must first configure FoolFuuka to use the Asagi database created in the previous steps. This can be done by following the steps listed below:

1. Access the FoolFuuka Administrative Panel
2. Navigate to `Preferences` under the `Boards` section
3. Set the `Boards Database` field to the same database name used in the `asagi.json` config file
4. Save your changes

Note: The steps listed above only need to be completed once.

In order to access the boards being archived with Asagi, you will need to add the boards to FoolFuuka by following the steps listed below:

1. Access the FoolFuuka Administrative Panel
2. Navigate to `Manage` under the `Boards` section
3. Click "Add Board"
4. Fill out the required fields properly
5. Check the "Is this an archived board?" checkbox
6. Click "Submit" to add the board to the database

Note: You will need to repeat the steps listed above each time you wish to add a board archived by Asagi.

5.2 Upgrade Guide

5.2.1 From 2.0 to 2.1

As of `2.1.0`, we've deprecated the original FoolFuuka installer repository and changed the folder structure of the FoolFuuka installations.

Backup

You will need to create a backup of the following folders or move them to a new location:

- `app/`
- `public/foolfuuka/boards/`

Note: Depending upon your configuration, you may not need to backup the `public/foolfuuka/boards/` directory.

Obtain the Latest Code

You can install the latest stable version of `2.1` by using one of the following methods:

Via Composer

```
$ composer create-project foolz/foolfuuka foolfuuka
$ cd foolfuuka
$ composer dump-autoload --optimize
```

Via Source Code

```
$ git clone https://github.com/FoolCode/FoolFuuka foolfuuka
$ cd foolfuuka
$ git checkout 2-1-stable
$ composer install --optimize
```

Copy App Files and Data

You will need to restore the following folders from the backup to the same locations:

- `app/`
- `public/foolfuuka/boards/`

Note: Depending upon your configuration, you may not need to restore the `public/foolfuuka/boards/` directory.

Developer Guide

6.1 Code Documentation

6.1.1 REST API

Index

GET `/_/api/chan/index/`

Property	Type	Description	Required
board	*string*	This is the shortname for the board.	Y
page	*integer*	The page number of the index.	Y

```
{
  "2": {
    "omitted": 10,
    "images_omitted": 10,
    "op": {
      "<POST OBJECT>"
    },
    "posts": [
      {
        "<POST OBJECT>"
      }
    ]
  },
  "1": {
    "omitted": 10,
    "images_omitted": 10,
    "op": {
      "<POST OBJECT>"
    },
    "posts": [
      {
        "<POST OBJECT>"
      }
    ]
  }
}
```

Search

GET `/_/api/chan/search/`

Property	Type	Description	Required
board	*mixed*	This is the shortname for the board.	Y
email	*string*		N
username	*string*		N
tripcode	*string*		N
capcode	*string*		N
subject	*string*		N
text	*string*		N
filename	*string*		N
filehash	*string*		N
deleted	*integer*		N
ghost	*integer*		N
filter	*integer*		N
date_start	*string*		N
date_end	*string*		N
order	*string*		N

```
[
  {
    "posts": [
      {
        "<POST OBJECT>"
      },
      {
        "<POST OBJECT>"
      }
    ]
  }
]
```

Thread

GET `/_/api/chan/thread/?board=dev&num=1`

Property	Type	Description	Required
board	*string*	This is the shortname for the board.	Y
num	*integer*	This is the post number of the thread.	Y
latest_doc_id	*integer*	This is the latest *doc_id* used as a starting point.	N
last_limit	*integer*	This limits the results to the last *x* posts.	N

```
{
  "1": {
    "op": {
      "<POST OBJECT>"
    },
    "posts": {
      "2": {
        "<POST OBJECT>"
      },
      "3": {
        "<POST OBJECT>"
      }
    }
```

```
    }
}
```

Post

GET `/_/api/chan/post/?board=dev&num=1`

Property	Type	Description	Required
board	*string*	This is the shortname for the board.	Y
num	*mixed*	This is the post number.	Y

```json
{
  "doc_id": "1",
  "poster_ip": "1111111111",
  "num": "1",
  "subnum": "0",
  "thread_num": "1",
  "op": "1",
  "timestamp": "1339024666",
  "timestamp_expired": "0",
  "capcode": "A",
  "email": null,
  "name": "Anonymous",
  "trip": null,
  "title": null,
  "comment": "COMMENT DATA HERE",
  "poster_hash": "fUSBgQ2y",
  "poster_country": null,
  "deleted": "0",
  "sticky": "0",
  "comment_processed": "COMMENT DATA HERE",
  "title_processed": "",
  "name_processed": "Anonymous",
  "email_processed": "",
  "trip_processed": "",
  "poster_hash_processed": "fUSBgQ2y",
  "fourchan_date": "6\/6\/12(Wed)23:17",
  "comment_sanitized": "COMMENT DATA HERE",
  "poster_country_name_processed": null,
  "media": {
    "op": "1",
    "media_id": "1024",
    "spoiler": "0",
    "preview_orig": "13390246665411s.jpg",
    "preview_w": "216",
    "preview_h": "250",
    "media_filename": "8211205.jpg",
    "media_w": "742",
    "media_h": "860",
    "media_size": "130990",
    "media_hash": "P2asAleYuUWVvEFBotaaxA==",
    "media_orig": "13390246665411.jpg",
    "exif": null,
    "total": "1",
    "banned": "0",
    "media": "13390246665411.jpg",
    "preview_op": "13390246665411s.jpg",
```

```
    "preview_reply": null,
    "media_status": "normal",
    "safe_media_hash": "P2asAleYuUWVvEFBotaaxA",
    "preview_orig_processed": "13390246665411s.jpg",
    "media_filename_processed": "8211205.jpg",
    "media_hash_processed": "P2asAleYuUWVvEFBotaaxA==",
    "media_link": "https:\/\/0-media-cdn.foolz.us\/ffuuka\/board\/dev\/image\/1339\/02\/1339024666541
    "remote_media_link": "https:\/\/0-media-cdn.foolz.us\/ffuuka\/board\/dev\/image\/1339\/02\/133902
    "thumb_link": "https:\/\/0-media-cdn.foolz.us\/ffuuka\/board\/dev\/thumb\/1339\/02\/1339024666541
  }
}
```

6.1.2 Software Hooks

Plugin

Foolz\Plugin\Plugin::execute#<plugin-name>

```
->setObject($plugin)
->setParam('context', $this->getContext())
```

Foolz\Foolframe\Model\Plugin::install#<plugin-name>

```
->setParam('context', $this->getContext())
->setParam('schema', $sm->getCodedSchema())
```

FoolFrame

Foolz\Foolframe\Controller\Admin::before#var.sidebar

```
->setObject($this)
->setParam('sidebar', [])
```

Foolz\Foolframe\Model\Context::handleConsole#obj.app

```
->setObject($this)
->setParam('application', $application)
```

Foolz\Foolframe\Model\Context::handleWeb#obj.afterAuth

```
->setObject($this)
->setParam('route_collection', $this->route_collection)
```

Foolz\Foolframe\Model\Context::handleWeb#obj.routing

```
->setObject($this)
->setParam('route_collection', $this->route_collection)
```

Foolz\Foolframe\Model\Context::handleWeb#obj.context

```
->setObject($this)
```

Foolz\Foolframe\Model\Context::handleWeb#obj.request

```
->setObject($this)
->setParam('request', $request)
```

Foolz\Foolframe\Model\Context::handleWeb#obj.response

```
->setObject($this)
->setParam('request', $request)
```

Foolz\Foolframe\Model\Preferences::load#var.preferences

```
->setObject($this)
->setParam('preferences', $this->preferences)
```

Foolz\Foolframe\Model\SchemaManager::forge#var.ignorePrefix

```
->setObject(new static())
->setParam('prefixes', $prefixes)
```

Foolz\Foolframe\Model\SchemaManager::forge#var.tables

```
->setObject(new static())
->setParam('tables', $tables)
```

Foolz\Foolframe\Model\System::getEnvironment#var.environment

```
->setParam('environment', $environment)
```

FoolFuuka

Foolz\Foolfuuka\Model\Comment::processComment#var.greentext

```
->setParam('html', $html)
```

Foolz\Foolfuuka\Model\Comment::processExternalLinks#var.link

```
->setObject($this)
->setParam('data', $data)
->setParam('build_href', $build_href)
```

Foolz\Foolfuuka\Model\Comment::processInternalLinks#var.link

```
->setObject($this)
->setParam('data', $data)
->setParam('build_url', $build_url)
```

Foolz\Foolfuuka\Model\CommentInsert::insert#obj.captcha

```
->setObject($this)
```

Foolz\Foolfuuka\Model\CommentInsert::insert#obj.afterInputCheck

```
->setObject($this)
```

Foolz\Foolfuuka\Model\CommentInsert::insert#obj.comment

```
->setObject($this)
```

Foolz\Foolfuuka\Model\Context::loadRoutes#obj.beforeRouting

```
->setObject($this)
->setParam('route_collection', $route_collection)
```

Foolz\Foolfuuka\Model\Context::loadRoutes#var.collection

```
->setParam('default_suffix', page)
->setParam('suffix', page)
->setParam('controller', 'Foolz\\Foolfuuka\\Controller\\Chan::*')
```

Foolz\Foolfuuka\Model\Context::loadRoutes#obj.afterRouting

```
->setObject($this)
->setParam('route_collection', $route_collection)
```

Foolz\Foolfuuka\Model\Media::getLink#exec.beforeMethod

```
->setObject($this)
->setParam('thumbnail', $thumbnail)
```

Foolz\Foolfuuka\Model\Media::insert#var.media

```
->setParam('dimensions', $dimensions)
->setParam('file', $file)
->setParam('name', $name
->setParam('path', $path)
->setParam('hash', $hash)
->setParam('size', $size)
->setParam('time', $time)
->setParam('media_orig', $media_orig)
->setParam('preview_orig', $preview_orig)
```

Foolz\Foolfuuka\Model\Media::insert#exec.createThumbnail

```
->setObject($this)
->setParam('is_op', $is_op)
->setParam('media', $media)
->setParam('thumb', $thumb)
->setParam('thumb_width', $thumb_width)
->setParam('thumb_height', $thumb_height)
->setParam('exec', $exec)
```

Foolz\Foolfuuka\Model\MediaFactory::forgeFromUpload#var.config

```
->setParam('ext_whitelist', [])
->setParam('mime_whitelist', [])
```

Foolz\Foolfuuka\Model\RadixCollection::structure#var.structure

```
->setParam('structure', $structure)
```

Foolz\Foolfuuka\Model\RadixCollection::preload#var.radixes

```
->setObject($this)
->setParam('preloaded_radixes', $this->preloaded_radixes)
```

6.2 Contribute

6.2.1 Coding Guidelines

FoolFuuka follows the *PSR-0* and *PSR-0* coding standards. In addition to these standards, the guidelines listed below must be followed as well:

- Function and Control Structure opening { MUST BE on a seperate line

6.2.2 IRC Channel

We have an IRC channel on the IRCHighWay network that can be used to discuss issues, feature requests, and various other related topics at #fooldriver.

6.2.3 Issue Tracker

You can find our issue tracker at our FoolFuuka GitHub repository.