

Pod Caster user guide

John-Paul Stanford <dev@stanwood.org.uk>

Pod Caster user guide

by John-Paul Stanford

Copyright © 2007 John-Paul Stanford

This document describes usage of the application Pod Caster version 1.0

Table of Contents

| | |
|---|---|
| 1. Introduction | 1 |
| Features | 1 |
| 2. Installation | 2 |
| 1. Linux Binary Distribution Packages | 2 |
| 2. Any platform static Jar Disturbution | 2 |
| 3. CLI Usage | 3 |
| podcaster | 3 |
| Optional arguemnts: | 3 |
| Required arguemnts: | 4 |
| Examples | 4 |
| capture-stream | 4 |
| Optional arguemnts: | 5 |
| Required arguemnts: | 5 |
| 4. Configuration | 7 |
| 5. Credits and Licenses | 8 |

List of Examples

| | |
|---|---|
| 3.1. Using podcaster in a cron job | 4 |
| 4.1. The default configuration file | 7 |

Chapter 1. Introduction

Pod Caster is a application which can capture Internet radio stations and store them locally. It can then add them to a pod cast for downloading onto portal devices. The encoded podcast/media files can have cover art/meta data added to them.

Features

- Capture Internet radio to audio files.
- Create MP3, WAV and MP4 files.
- Encodes metadata and cover art into the media files.
- Create pod cast from captured radio stations.
- Supports real audio and Microsoft ASF format.

Chapter 2. Installation

There are a few different ways to install Pod Caster. The different distributions can be found at *Downloads* [<http://code.google.com/p/pod-caster/downloads/list>]

All of the distributions will require a Java 1.6 compatible JRE.

1. Linux Binary Distribution Packages

There are several Linux distribution packages that can be found in the downloads section of the website. If your distribution uses RPM packages, then following these instructions.

1. Pick the packages for your distribution
2. Download them to a directory
3. Change to that directory from the console
4. Log in as root and run the command **rpm -Uvh *.rpm**

This will install the application and scripts to launch it. This will install the scripts **/usr/bin/capture-stream** and **/usr/bin/podcaster**.

2. Any platform static Jar Distribution

There is a static jar distribution available on the downloads page. This will run on any platform with the correct JRE. Simply download it to your system, and type the command:

```
java -jar podcaster-1.0-static-bin.jar [options]
```

This can be used to capture audio and add it to a podcast the same command line parameters as the podcaster script.

Chapter 3. CLI Usage

podcaster

The podcaster command is used capture audio from internet radio stations and store it as a audio file in a directory. This audio file is then added to a podcast.

```
podcaster [-h] [OPTIONS...] -f -i -rf -ru -t -u
```

This command uses the system tools **ffmpeg** and **mplayer**, by default the tool will look for these in /usr/bin, however it possible to change the location of these via a configuration file. See the configuration chapter for more information on the format.

Optional arguemnts:

-a, --metaArtworkUrl

The URL to the cover artwork that should be encoding into the captured audio file. If this option is missing, the audio file will not have any cover artwork.

-c, --metaCopyright

If this option is given, then a copyright message is stored in the captured audio file.

-ed, --entryDescription

If this option is given, then a description is encoded as metadata in the captured audio file

-fa, --feedArtwork

This option is used to get a URL which points to cover artwork that can be used as the feeds artwork image. The artwork will be downloaded to file and placed next to the feed on the filesystem. If the feed already exsits, then this option tells it to use this artwork instead of any artwork that might have been used in the past.

-fd, --feedDescription

This option will place a description used to describe the podcast into the rss feed. If the feed already exsits, then this option tells it to use this description instead of any description that might have been used in the past.

-l, --log_config

This option is used to control the loggin of the tool. The default option if this option is missing is *INFO*, which will log only messages considered suiteable for information level. If the option *DEBUG* is used then their will also be debug messages and the format will be changed to make debugging easier.

It's also possible to pass a log4j configuration file path to this option which will cause the logging setup to be configured with the contents of the file.

-m,--maxEntries

Used to set the maximum number of entries in the podcast rss feed. If the number in the feed exceeds the amount given, then entries are deleted, starting with the oldest until there is the same number in the feed as the number given with this option. If entries are removed, then any files they reference are also deleted.

-r, --metaArtist

This option is used to encode the artist into the audio files metadata and it's also used by the podcast rss feed.

-h, --help

Show the help message

-c, --config_file

This option is used to override the default configuration file been used.

Required arguments:

-f, --format

Used to set the format of the audio file is saved in. This can be either WAV, MP3 or MP4

-i, --metaTitle

This option is used to set the title of the entry

-rf, --rssFile

This is used to set the path to the podcast RSS file. This is the location the RSS file is saved and updated.

-ru, --rssUrl

This is used to set a webserver URL to the RSS feed, this is encoded into the podcast RSS feed.

-t, --time

This is used to set the duration the audio should be captured for. The value is in milliseconds

-u, --url

This is used to supply the URL to the audio stream that is to be captured

Examples

Example 3.1. Using podcaster in a cron job

The following line is an example of using podcaster to record a radio show and insert it into a podcast. This will record the Z
`<code> 50 18 * * 1 /usr/bin/podcaster --metaArtworkUrl="http://node1.bbcimg.co.uk/iplayer/images/episode/b00lgm79_6"`

capture-stream

The capture-stream command is used capture audio from internet radio stations and store it as a audio file. It will also store metadata passed on the command line in the file if the format allows it.

```
capture-stream [-h] [OPTIONS...] -u -t -o
```

This command uses the system tools **ffmpeg** and **mplayer**, by default the tool will look for these in /usr/bin, however it possible to change the location of these via a configuration file. See the configuration chapter for more information on the format.

Optional arguemnts:

-f, --format

Used to set the format of the audio file is saved in. This can be either WAV, MP3 or MP4

-i, --metaTitle

This option is used to set the title in the metadata of the caputred audio file

-a, --metaArtworkUrl

A URL to a artwork image which will be saved in the caputred audio file as it's artwork

-c, --metaCopyright

If this option is given, then a copyright message is stored in the captured audio file.

-r, --metaArtist

This option is used to set the artist in the metadata of the caputred audio file

-e, --metaDescription

This option is used to set the comment in the metadata of the caputred audio file

-l, --log_config

This option is used to control the loggin of the tool. The default option if this option is missing is *INFO*, which will log only messages considered suiteable for information level. If the option *DEBUG* is used then their will also be debug messages and the format will be changed to make debugging easier.

It's also possible to pass a log4j configuration file path to this option which will cause the logging setup to be configured with the contents of the file.

-c, --config_file

This option is used to override the default configuration file been used.

Required arguemnts:

-u, --url

This is used to supply the URL to the audio stream that is to be captured

-o, --output

The path to the file the captured audio should be saved in.

-t, --time

This is used to set the duration the audio should be captured for. The value is in milliseconds

Chapter 4. Configuration

The applications make use of a XML configuration file. This tells them which for example were to find the system tools like mplayer and ffmpeg

The applications have CLI options that can be used too tell it which configuration file to use. If this option is not present, then it will look for the file at the location `/etc/podcaster-conf.xml`. If this can't be found, it will fall back too using default settings.

Example 4.1. The default configuration file

```
<config><sources><!-- www.tv.com TV Show information source --><source id="org.sta
```

Chapter 5. Credits and Licenses

Pod Caster

- John-Paul Stanford <dev@stanwood.org.uk> - Original Author

Documentation copyright 2008-2009, John-Paul Stanford <dev@stanwood.org.uk>