tvrenamr Documentation

Release 3.0

2011, George Hickman

February 06, 2012

CONTENTS

A command line utility to rename TV show files into a cleaner format.

CONTENTS

1.1 Installation

1.1.1 Requirements

- Python 2.6+
- PyYaml
- The Interwebs

1.1.2 Python Package Index

Install Tv Renamr with your favourite Python package manager:

```
pip install tvrenamr
```

Note: You might need to be an administrator to do this.

A nice shiny `tvr` script is now installed in the relevant directory on your python path.

Note: If you don't have pip install you'll get an error message. Instructions on how to install pip can be found here.

Windows

Due to a problem with easy_install/pip and Tv Renamr's option parsing code it isn't possible to use easy_install or pip for installation just yet, but it might get fixed in the future.

1.1.3 Source: GitHub

```
git clone https://github.com/ghickman/tvrenamr.git
cd tvrenamr
python setup.py install
```

> **Warning:** You may probably need to be an administrator to run the `python setup.py install` line.

1.2 Usage

In your favourite terminal run:

```
$ tvr file
```

or

```
$ tvr folder
```

1.2.1 Options

Tv Renamr accepts the following options which take precedent over any options set in the *Config*.

-c, --canonical	Set the show's canonical name to use when performing the online lookup.
--deluge	Checks Deluge to make sure the file has been completed before renaming.
--deluge-ratio	Checks Deluge for completed and that the file has at least reached X share ratio.
-d, --dry-run	Dry run your renaming.
-e, --episode	Set the episode number. Currently this will cause errors when working with more than one file.
--log-file	Set the log file location.
-l, --log-level	Set the log level. Options: short, minimal, info and debug.
--library	Set the library to use for retrieving episode titles. Options: thetvdb & tvrage.
-n, --name	Set the show's name. This will be used as the show's when the renaming is completed.
-o, --output	Set the output format for the episodes being renamed.
--organise	Organise renamed files into folders based on their show name and season number.
--no-organise	Explicitly tell Tv Renamr not to organise renamed files. Used to override the config.
-q, --quiet	Don't output logs to the command line.
-r, --recursive	Recursively lookup files in a given directory.
--rename-dir	The directory to move renamed files to, if not specified the working directory is used.
--no-rename-dir	Explicitly tell Tv Renamr not to move renamed files. Used to override the config.
--regex	The regular expression to use when extracting information from files.
-s, --season	Set the season number.
-t, --the	Set the position of 'The' in a show's name to the end of the file.

Examples

```
$ tvr
```

```
$ tvr /path/to/a/directory/
```

```
$ tvr /path/to/a/file.avi

$ tvr --organise -r '/path/to/a/directory/' /path/to/a/directory/[a_file.avi]

$ tvr --season '1' --name 'chuck' /path/to/a/file/the_file.avi
```

1.3 Config

For ease of use Tv Renamr uses a config file. By default it looks for it in ~/.tvrenamr/config.yml, an example
is listed *below* which shows all the possible default values you can use.

1.3.1 Defaults

The defaults segment should be self-explanatory but I'll list them just for completeness. The listed values are the
assumed defaults if any of the options are not added.

Format

The output format for files to be renamed to.

```
format: '%n - %s%e - %t%x'
```

- %n: Show name
- %s: Season Number
- %e: Episode Number
- %t: Episode Title
- %x: Extension

Note: The extension part includes the period (.) part of the file's extension and is also optional.

Library

The online database to use for episode names. Options are: thetvdb or tvrage

```
library: thetvdb
```

Organise

Organise your files within the renamed directory.

```
organise: yes
```

Renamed

The directory to move your renamed files to.

```
renamed: /Volumes/Media/TV/
```

The

If a show has a leading 'The', such as 'The Big Bang Theory', move it to the end of the show name, i.e. 'Big Bang Theory, The'.

```
the: true
```

1.3.2 Tv Shows

Below the defaults are shows that won't get renamed correctly using the default options. Taking CSI as the example you have:

```
csi:
    canonical: "CSI: Crime Scene Investigation"
    output: "CSI, Crime Scene Investigation"
```

Show Name

In the above example `csi` is used to match the show name in the downloaded file name, which might look something like this `csi.s10e01.blah.blah.avi`.

Canonical

The name used by the online database(s) for a show.

Since The TVDb and Tv Rage both list CSI as *CSI: Crime Scene Investigation* the `canonical` option is used.

```
canonical: "CSI: Crime Scene Investigation"
```

Note: This method is the easiest way to deal with shows with a year in the name too, i.e. Castle (2009).

Output

The show name to use when writing the new filename.

The canonical show name contains a colon which most filesystems won't play nice with

```
output: "CSI, Crime Scene Investigation"
```

Format

The output format to use when writing the new filename.

```
format: %n - %s%e
```

> **Warning:** The colon (`:`) and comma (`,`) characters are reserved in YAML so must be quoted.

```
defaults:
  format: '%n - %s%e - %t%x'
  library: thetvdb
  organise: yes
  renamed: /Volumes/Media/TV/
  the: true

'24':
  format: '%n - %s%e'

american dad:
  canonical: American Dad!

castle 2009:
  canonical: Castle (2009)

csi:
  canonical: "CSI: Crime Scene Investigation"
  output: "CSI, Crime Scene Investigation"

doctor who 2005:
  canonical: Doctor Who (2005)

the it crowd:
  the: false

the simpsons:
  the: false

v 2009:
  canonical: V (2009)
  output: V
```

1.4 Logging and Command Line Output

The logging level for the log file is always `debug`, however the amount of information you see in your console can be modified. Use the `--log-level` option (`-l` for short) to select one of the output options:

- short
- minimal
- info
- debug

Given the filename `chuck.S01E02.avi` to rename, the log levels would show the following information:

1.4.1 Short

> Renamed: "Chuck - 102 - Chuck Versus the Helicopter.avi"

1.4.2 Minimal

Renaming: chuck.S01E02.avi

Directory: /Volumes/Media/TV/Chuck/Season 1/

Renamed: "Chuck - 102 - Chuck Versus the Helicopter.avi"

1.4.3 Info

Renaming: chuck.S01E02.avi

Searching: chuck

Episode: Chuck Versus the Helicopter

Directory: /Volumes/Media/TV/Chuck/Season 1/

Renamed: "Chuck - 102 - Chuck Versus the Helicopter.avi"

1.4.4 Debug

2010-10-17 20:40 DEBUG Config Config loaded

2010-10-17 20:40 DEBUG Config Defaults retrieved

2010-10-17 20:40 SHORT Core Dry Run beginning.

2010-10-17 20:40 SHORT Core ——

2010-10-17 20:40 SHORT Core

2010-10-17 20:40 MINIMAL Core Renaming: chuck.S01E02.avi

2010-10-17 20:40 DEBUG Core Renaming using:
(?P<show>[ws.,_-]+).[Ss]?(?P<season>[d]{1,2})[XxEe]?(?P<episode>[d]{2})

2010-10-17 20:40 DEBUG Core Returned show: chuck, season: 01, episode: 02, extension: .avi

2010-10-17 20:40 DEBUG Core Imported The Tv Db library

2010-10-17 20:40 INFO The Tv DB Searching: chuck

2010-10-17 20:40 DEBUG The Tv DB Retrieving series id for chuck

2010-10-17 20:40 DEBUG The Tv DB Series url:
http://www.thetvdb.com/api/GetSeries.php?seriesname=chuck

2010-10-17 20:40 DEBUG The Tv DB XML: Attempting to parse

2010-10-17 20:40 DEBUG The Tv DB XML retrieved, searching for series

2010-10-17 20:40 DEBUG The Tv DB Series chosen: Chuck

2010-10-17 20:40 DEBUG The Tv DB Retrieved show id: 80348

2010-10-17 20:40 DEBUG The Tv DB Retrieved canonical show name: Chuck

2010-10-17 20:40 DEBUG The Tv DB Episode URL:
http://www.thetvdb.com/api/C4C424B4E9137AFD/series/80348/default/1/2/en.xml

2010-10-17 20:40 DEBUG The Tv DB Attempting to retrieve episode name

2010-10-17 20:40 DEBUG The Tv DB XML: Retreived

2010-10-17 20:40 DEBUG The Tv DB XML: Attempting to parse

2010-10-17 20:40 DEBUG The Tv DB XML: Parsed

2010-10-17 20:40 DEBUG The Tv DB XML: Episode document retrieved for Chuck - 0102

2010-10-17 20:40 DEBUG The Tv DB XML: Attempting to finding the episode name

2010-10-17 20:40 DEBUG The Tv DB Retrieved episode name: Chuck Versus the Helicopter

2010-10-17 20:40 INFO Core Episode: Chuck Versus the Helicopter

2010-10-17 20:40 DEBUG Error 'chuck' is not in the Config. Falling back on name extracted from the filename

2010-10-17 20:40 DEBUG Core Using the formatted show name retrieved by the library: Chuck

2010-10-17 20:40 DEBUG Core Final show name: Chuck

2010-10-17 20:40 MINIMAL Core Directory: /Volumes/Media/TV/Chuck/Season 1/

2010-10-17 20:40 DEBUG Core Full path: /Volumes/Media/TV/Chuck/Season 1/Chuck - 102 - Chuck Versus the Helicopter.avi

2010-10-17 20:40 MINIMAL Core Renamed: "Chuck - 102 - Chuck Versus the Helicopter.avi"

2010-10-17 20:40 SHORT Core

2010-10-17 20:40 SHORT Core ———————————————————————————————————–

2010-10-17 20:40 SHORT Core Dry Run complete. No files were harmed in the process.

2010-10-17 20:40 SHORT Core

Debug will automatically perform a dry run rename since it was designed to be used for testing. However this is the format you will see in your log file, minus the dry run.

1.5 Custom Regular Expressions

By default TV Renamr will match shows in the formats:

- show.s0e00

- show.0x00

However you can specify custom regular expressions if your files aren't in either of these formats. Some custom regular expression syntax has been used to help you specify different parts of the filename:

- Show: `%n` - `(?P<show>[\w\s.,_-]+)`

- Season: `%s` - `(?P<season>[\d]{1,2})`

- Episode: `%e` - `(?P<episode>[\d]{2})`

It is also possible to specify how many digits there are in the season and episode sections of the filename using the syntax:

- Season: `%s{n}`

- Episode: `%e{n}`

where the n in `{n}` specifies how many digits are in each of the sections.

Note: All spaces are converted to periods before your regular expression is run.

Python regular expression syntax can be found here.

1.6 Custom Output

It is possible to set a custom output format for your files using the `-o` command line option and the same syntax used with *Custom Regular Expressions*:

- `%n` - Show

- `%s` - Season

- `%e` - Episode

- `%t` - Title

- `%x` - Extension

1.6.1 Example

```
$ tvr -o "%s%e - %n - %t%x" chuck.S01E02.avi
```

Note: Not including the `%x` section on Windows systems can cause problems when trying to run your media files.

The `-o` option is equivalent to the `format` option in the *Config* which can be set for a show or in the defaults section.

1.7 Known Issues

Please report any bugs or feature requests to the issue tracker on Github.

1.7.1 Colons

All colons `:` are converted to commas `,` in both the show name and the episode title. This stops issues that can arise with network shares and gives a cleaner format on windows where python replaces the colon character with a backslash `\`.

1.7.2 Years in Show Names

Shows with a year in the canonical title, i.e. Doctor Who (2005), will fail if this year isn't specified in the show name inside brackets. You'll need to set this as the canonical name in the config file.

1.7.3 Multiple Episodes

Multiple episode files are just not coped with at all. The first episode is usually picked up so the file will be renamed using this.

1.8 Changelog

1.8.1 v3.0.2

- Fix a silly naming bug that broke everything and brown bagged the last release.

1.8.2 v3.0.1

- Allow the use of apostrophes in the show name regular expression
- Allow the use of hypen as a delimiter in the filename regular expression.
- Clean up the interface to the episode object for use in the front end.
- Use the correct variable name when retrieving the show name from an episode object.
- Give more sensible output when the config's defaults are missing.

- Return the correct error code when exiting from an error.

1.8.3 v3.0.0

- 720p episodes can now be renamed.

- Fallback to the other library if the first one can't find a tv show or episode. This feature also adds better support for new libraries.

- Added a command line option to override the show name when output to the filename.

- Can use foward slashes allowed in show names and episode titles.

- Use lxml as the xml library and add an extra check for empty xml files being returned from the library.

- Use an episode object to hold an episode's information during the rename process.