webtoolbox Documentation

Release 1.0.0

Chris Adams

Oct 24, 2017

Contents

CHAPTER 1

Installation

This project assumes the use of pip, virtualenv and virtualenvwrapper. If you don't already have them:

```
easy_install pip
pip install virtualenv virtualenvwrapper
. virtualenvwrapper_bashrc
mkdir ~/.virtualenvs
```

Once they're setup you'll want to create a virtualenv:

```
mkvirtualenv webtoolbox
add2virtualenv /path/to/webtoolbox
```

Now you're ready to install our prerequisites:

```
pip install -r requirements.pip
```

Note: Tornado uses `pycurl`, which may or may not install correctly on a Mac using a simple `pip install pycurl`. If you encounter problems follow the instructions in the Tornado documentation.

To use the redbot-based tools. This is complicated by the fact that redbot hasn't been turned into an importable module yet:

```
pip install -e git://github.com/mnot/nbhttp.git@master#egg=nbhttp
git clone http://github.com/mnot/redbot
add2virtualenv redbot/src
```

CHAPTER 2

The Tools

Spiders

check_site

synopsis Site validation spider

A site validator which uses `webtoolbox.clients.Spider` to process an entire site and checking for bad links, 404s, and optionally HTML validation. It generates either text or HTML reports and can be used to generate lists of site URLs for use with load-testing tools like http_bench or *wk_bench*.

--help
Display all available options and full help

-v

--verbosity
Increase the amount of information displayed or logged

--validate-html
Process all HTML using HTML Tidy and report any validation errors

--format=REPORT_FORMAT
Generate the report as HTML or text

--report=REPORT_FILE
Save report to a file instead of stdout

--skip-media
Skip media files: <img>, <object>, etc.

--skip-resources
Skip resources: <script>, <link>

--skip-link-re=SKIP_LINK_RE
Skip links whose URL matches the specified regular expression

--save-page-list=PAGE_LIST
Save a list of URLs for HTML pages in the specified file for use with a tool like http_bench or *wk_bench*

--save-resource-list=RESOURCE_LIST
Save a list of URLs for pages resources in the specified file

--log=LOG_FILE
Specify a location other than stderr

--simultaneous-connections=2
Adjust the number of simultaneous connections which will be opened to the server

red_spider

synopis Site validation spider based on `redbot`

Mark Nottingham released redbot - a modern replacement for the classic cacheability tester. I've been using it at work to audit website performance before releases since proper HTTP caching makes an enormous difference in perceived site performance.

redbot is a focused tool and provides a great deal of detail about at most one page and, optionally, its resources. I wanted to expand the scope to testing an entire site and performing content validation and created **red_spider.py** which allows you to perform all of those checks by spidering an entire site, receiving a nice HTML report and, optionally, also validating page contents as well.

--help
Display all available options and full help

--format=REPORT_FORMAT
Generate the report as HTML or text

--report=REPORT_FILE
Save report to a file instead of stdout

--validate-html
Validate HTML using tidylib

--skip-media
Skip media files: <img>, <object>, etc.

--skip-resources
Skip resources: <script>, <link>

--skip-link-re=SKIP_LINK_RE
Skip links whose URL matches the specified regular expression

--save-page-list=PAGE_LIST
Save a list of URLs for HTML pages in the specified file

--save-resource-list=RESOURCE_LIST
Save a list of URLs for pages resources in the specified file

--log=LOG_FILE
Specify a location other than stderr

-v

--verbosity

Increase the amount of information displayed or logged

Load Generators

wk_bench

synopsis Benchmark user-perceived page time for a list of URLs using a true WebKit browser

Mac OS X-specific tool which uses PyObjC to load pages in WebKit. Takes URLs on the command-line or in a separate file and runs through them as quickly as possible, measuring the time it takes from beginning the request until the browser fires the `didFinishLoadForFrame` event, which includes things like image loading, Flash, JavaScript, etc. for measuring user-perceptible page-load performance.

`--help`
Display all available options and full help

log_replay

synopsis Replay webserver log files in realtime

If you need to replace webserver log files at something approximating realtime, **`log_replay`** is your friend. It uses Tornado's non-blocking HTTP client to fetch all of the URLs but will sleep any time it's too far ahead of the simulated virtual time.

`--help`
Display all available options and full help

CHAPTER 3

Python Modules

Clients

Indices and tables

- genindex
- modindex
- search

Index

Symbols

C

L

R

W