1 Overview

unisos.wsf: Somewhat General purpose Web Scraping Framework (WSF)

2 Support

For support, criticism, comments and questions; please contact the author/maintainer
Mohsen Banan at: http://mohsen.1.banan.byname.net/contact

3 Documentation

Theree is no silver bullet for universally providing web scraping capabilities.

Each web site is different and various technologies, encodings and paradigms are used.

What can be done is to create a framework that addresses some of the common aspects of this problem domain.

This is what Web Scraping Framework (WSF) tries to do.

Common aspects of Web scraping include:

Configuration: WSF uses python function invocation as the configuration syntax.

```
See ./unisos/wsf/wsf_config.py for details.
```

Simultaneous Parallel Dispatch Of Multiple Urls: WSF provides a rudimentary mechanism for parallel dispatch.

```
See ./unisos/wsf/wsf_parallelProc.py for details.
```

Use of external systems such as celery for workers dispatch is a better solution for larger systems.

Retrieving Web Content As html: WSF uses requests to obtain html.

```
See ./unisos/wsf/wsf_inputs.py for details.
```

Digesting html: WSF uses Beautiful Soup 4 to digest html.

```
See ./unisos/wsf/wsf_digestHtml.py for details.
```

Basic Scraping Facilities: WSF provides some generic facilities for basic scripting as an abstract class.

```
See ./unisos/wsf/wsf_scraperBasic.py for details.
```

The ScraperBasic is an abstract and incomplete class which must be subclassed to become concrete.

Features provided by this principal class are:

- Capturing of Config parameters.
- Facilities for simple state transition.
- Facilities for maintaining results.

Multipage Scraping Facilities: WSF provides some facilities for web information that has been paginated.

```
See ./unisos/wsf/wsf_scraperMultipage.py for details.
```

The ScraperMultipage(wsf_scraperBasic.ScraperBasic) is an abstract and incomplete class which must be subclassed to become concrete.

ScraperMultipage itself is a subclass of wsf_scraperBasic.ScraperBasic.

Capturing Results And Writing Results: WSF provides a rudimentary mechanism for capturing intermediate results and their output.

```
See ./unisos/wsf/wsf_results.py for details.
```

Command Line Mapping: WSF can be used in combination with unisos.icm (Interactive Command Modules)

ICM can be thought of as a supper set of click which supports plugins as "load" parameters.

Both config files and concrete scraper classes can be passed to ICMs as "load" parameters.

4 Installation

4.1 From PyPi

```
pip install unisos.wsf
```

4.2 From File System

Go to the wsf/py3 directory.

```
Run: ./setup.py sdist
```

```
Run: pip install --no-cache-dir ./dist/unisos.wsf-0.1.tar.gz
```

5 Usage

```
import unisos.wsf
```

Use of unisos.wsf involves creating concrete subclasses of the set of abstract classes that wsf provides.