Crawler Framework - System Analyse

Aurther: Bryant Liu
Beginning Date: 2021/10/04

Language: Python 3
Package name: smoothcrawler
Description: Build a crawler program humanly.

Requirements

5W1H

For who:
Python developers

In where:
Python environment.

At (in) when:
Data mining or getting some data from the internet by web or app, etc.

Do what:
Internet Crawler.

How:
It could build web crawlers with different roles.
For instance, SimpleCrawler

Sample code:
result = SimpleCrawler.run(url=”something.com”)

ConcurrentCrawler

Sample code:
result = ConcurrentCrawler.run(url=”something.com”, exes=10)

Procedure Description and design

Procedure (first version - brief):

1. Initial something you need to prepare for the crawling task.
2. Send HTTP requests to the target server.
3. Get the HTTP response.
4. Do the data-handling process.
5. Do something after the data-handling process.
6. Shutdown the crawler.

Add detail description of procedure ():

1. Initial something you need to prepare for the crawling task.
   Do something indeed initialization before crawling data. For example, preparing target URLs.

2. Send HTTP requests to the target server.
   In general, it should provide different methods to use -- it means HTTP method, no matter which packages or libraries you choose to use.
   I include all features or parameters of the HTTP method.

3. Get the HTTP response.
   First of all, you should make sure that the status code is ‘200’ so that you could do any data-handling process. However, something unexpected always happens on the internet. Like it would return an error with status code ‘426’ because of too many requests in a short time period. Program will interrupt if you don’t handle it.

4. Do the data-handling process.
   Receive the result object from HTTP response.

5. Do something after the data-handling process.
   The most common scenario at this step is persistence. Saving the data which has been handled as one specific file format or saving it into a database.

6. Shutdown the crawler.
   It should close and join the workers if it runs tasks with multiple workers strategy.

Add more detail description and code implementation of procedure ():

1. Initial something you need to prepare for the crawling task.
   Do something indeed initialization before crawling data. For example, preparing target URLs.
   1-1. URL object:

   Exist reason:
   Sometimes, we need to crawl data with a time period like 1 year. But the API has a parameter date or datetime and the value like ‘20210101’ or ‘20210101000000’. We would need to re-send HTTP requests by an almost same URL with a loop to do it.

Example code

```
for i in range(1,365):
    url = ‘http://www.test.com?day=’ + i   # Just an example URL, it’s not real.
    response = send_http(url)
    …
```

So I think about this: is it possible to control it by an object?

Which rule does it need to have?

- By datetime (how to set the rule to loop?)
- By Unix time (for evaluating)
- By index
- By the target iterable object

Rules	Example Code

Datetime	# http://www.test.com/index?date=20210101
	url = URL(url=“http://www.test.com/index?date={date}”, rule=datetime_iter)

Unix Time (for evaluating)	# http://www.test.com/index?time=189784574
	url = URL(url=“http://www.test.com/index?time={time}”, rule=datetime_iter)

Index	# http://www.test.com/index?index=20
	url = URL(url=“http://www.test.com/index?date={index}”, rule=index_iter)

Iterable object	# http://www.test.com/index?param=test_1
	Iterator = [“test_1”, “test_2”, “test_3”]
	url = URL(url=“http://www.test.com/index?param={iterator}”, rule=iterator)

What attribute does it have?
- Base URL characters
-

What operators could it use? (No)
- Add (+)
- Decrease (-)
- Thinking: I think this idea is not necessary because it's not clear and intuitive.

How to use it?
- Pass it into the crawler via argument.

Example Code

# Method 1 	
url = URL(url="http://www.test.com/index?date={date}", rule=datetime_iter)
Crawler.run(url=url)   # Receive URL object
 	
# Method 2 	
Crawler.run(url="http://www.test.com/index?date=20210101")   # Overload of 'run' and receive string type URL parameter

1-2. Crawler object usage thinking:

2. Send HTTP requests to the target server.
   In general, it should provide different methods to use -- it means HTTP method, no matter which packages or libraries you choose to use.
   I include all features or parameters of the HTTP method.
   By the way, it should retry sending HTTP requests again if it gets a failed HTTP response (define successful response is status code 200 here). It even could sleep for several seconds or longer if in need.
   In Python, it could implement this feature with the package 'urllib' or 'requests'. It's free to developers which one you want to use. It would be an interface.

   2-1. For the HTTP requests handler, I consider that it could open 6 APIs (HTTP methods) outside: GET, POST, PUT, DELETE, HEAD, OPTION. However it also could only open 1 API outside and let developers decide which HTTP method they want to use and which parameters it needs to pass.

Example Code	Comments
class HTTPIO:	Thinking: Is it necessary to divide every HTTP method into a function?

`def get(self, *args, **kwargs):`
`    # Sending HTTP requests by GET implementation`
`    pass`
 	
`def post(self, *args, **kwargs):`
`    # Sending HTTP requests by POST implementation`
`    pass`
 	
`def put(self, *args, **kwargs):`
`    # Sending HTTP requests by PUT implementation`
`    pass`
 	
`def delete(self, *args, **kwargs):`
`    # Sending HTTP requests by DELETE implementation`
`    pass`
 	
`def head(self, *args, **kwargs):`
`    # Sending HTTP requests by HEAD implementation`
`    pass`
 	
`def option(self, *args, **kwargs):`
`    # Sending HTTP requests by OPTION implementation`
`    pass`
	Benefits:
	Dividing different HTTP methods logic so it also could do their own process with one specific HTTP method.
	
	Drawbacks:
	It’s a little bit difficult to use and maintain, and it also adds some complexity when developers extend it.

`class HTTPIO:`
 	
`    def request(*args, **kwargs):`
`        # Implement sending HTTP request`
`        pass`
	Thinking: This way will be clearer and cleaner than the previous one. But how about developers wanting to send multiple requests with different HTTP methods?
	
	Benefits:
	It’s so simple and focuses on one logic: sending HTTP requests.
	
	Drawbacks:
	Because it's simple, it can't handle different scenarios. In other words, developers still need to do some conditions to determine which HTTP method it is right now and what thing it needs to do with its HTTP method.

Above all, I consider third way to do:

It only opens 1 API outside (public), developers could implement something in it and call it. However, it also opens other 6 APIs “outside” (public or protected) to let developers to extend its features.

Example Code	Comments

class HTTPIO:
 	
    def request(method=GET, *args, **kwargs):
        If method == GET:
            self.get(*args, **kwargs)
        elif method == POST:
            self.post(*args, **kwargs)
        elif method == PUT:
            self.put(*args, **kwargs)
        elif method == DELETE:
            self.delete(*args, **kwargs)
        elif method == HEAD:
            self.head(*args, **kwargs)
        elif method == OPTION:
            self.option(*args, **kwargs)
        else:
            raise HTTPMethodError
 	
    def get(self, *args, **kwargs):
        # Sending HTTP requests by GET implementation
        pass
 	
    def post(self, *args, **kwargs):
        # Sending HTTP requests by POST implementation
        pass
 	
    def put(self, *args, **kwargs):
        # Sending HTTP requests by PUT implementation
        pass
 	
    def delete(self, *args, **kwargs):
        # Sending HTTP requests by DELETE implementation
        pass
 	
    def head(self, *args, **kwargs):
        # Sending HTTP requests by HEAD implementation
        pass
 	
    def option(self, *args, **kwargs):
        # Sending HTTP requests by
	Thinking: It opens 2 types APIs outsides: one is ‘request’ and another one is ‘request by HTTP method’.
	API ‘request’ has a parameter *method* to determine which HTTP method it needs to use (which methods it will call). Therefore, developers could override method ‘request’ to implement logic directly or they also could override the one specific function which mapping HTTP method.
	
	Benefits:
	Developers have multiple different ways to choose. They could use the best way or implementation to reach their target. It's cleaner and maintainer because different logics with HTTP methods are divided.
	
	Drawbacks:
	It may be more complex than previous 2 ways.

OPTION implementation pass	

3. Get the HTTP response.
   First of all, you should make sure that the status code is ‘200’ so that you could do any data-handling process. However, something unexpected always happens on the internet. Like it would return an error with status code ‘426’ because of too many requests in a short time period. Program will interrupt if you don’t handle it.
   There are so many HTTP status codes (1XX, 2XX, 3XX, 4XX and 5XX), I’m thinking how could I design here to let developers be more clear and familiar with it.

   3-1. Open all logic outside to let developers to implement.

Example Code:

```
@abstracted
def send_http():
    // Some implementations about sending HTTP requests.
```

   3-2. Annotation of some logic as functions to let developers to extend to overwrite.

Example Code:

```
def send_http_request():
    status_code = get_status_code()
    if status_code == 200:
        status_code_200()
    elif status_code == 201:
        status_code_201()

    ....

    elif status_code == 5XX:
        status_code_5XX()


def status_code_200():
    // Implementation about getting 200 HTTP responses.
    // Developers could overwrite this function if they needed to do something when
they got 200.


def status_code_201():
    // Implementation about getting 201 HTTP responses.
    // Developers could overwrite this function if they needed to do something when
they got 201.
```

```
...


def status_code_5XX():
    // Implementation about getting 5XX HTTP responses.
    // Developers could overwrite this function if they needed to do something when
they got 5XX.
```

The method naming collections below:

Methods naming	Comments

`def http_200_response(self):`
`    pass`	

`def status_code_200(self):`
`    pass`	

`def http_status_code_200(self):`
`    pass`	

`def status_code_200_response(self):`
`    pass`	

`def http_status_code_200_response(self):`
`    pass`	

`def http_sc_200_resp(self):`
`    pass`	

4. Do the data-handling process.
   Receive the result object from HTTP response. In general, the process receives a HTTPResponse object and we could get the text content, HTML or something else to do something we want to do.

5. Do something after the data-handling process.
   The most common scenario at this step is persistence. Saving the data which has been handled as one specific file format or saving it into a database.

   5-1. For this process, it could use 2 scenarios: ‘one connection with Lock’ and ‘multiple connection with Semaphore’.

   5-1-1. One connection with Lock

   5-1-2. Multiple connections with Semaphore

5-2. For persistence as a file(s) in this process, it could consider 4 scenarios: ‘one thread saving one file’, ‘all threads save as one file’, ‘one thread saving one file and all files in one compress file’, ‘multiple threads save as multiple files and all in one compress file’.

5-2-1. One thread saving one file

5-2-2. All threads saving one file

5-2-3. One thread saving one file and all files compress in one file

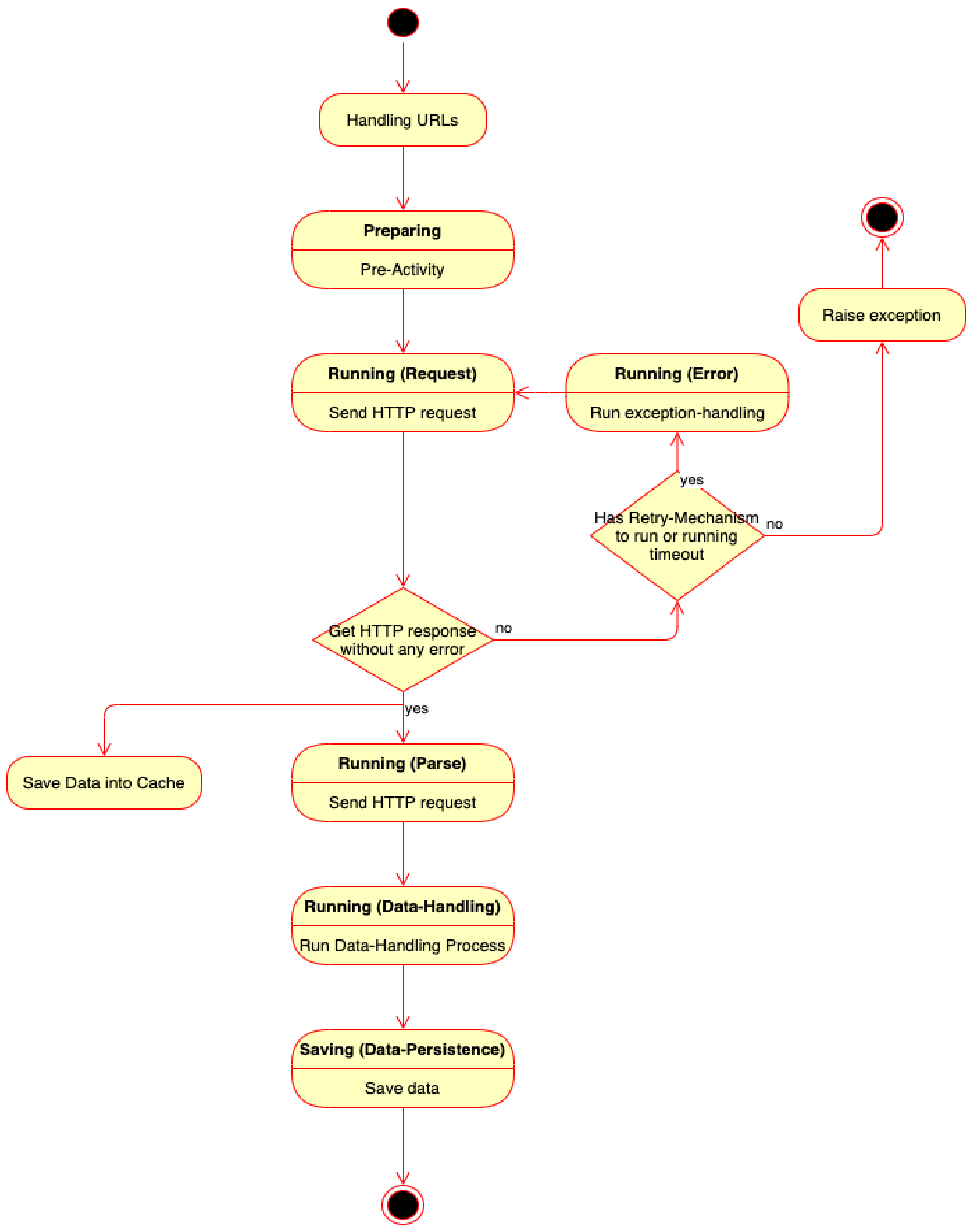
5-2-4. Multiple threads save as multiple files and all compress in one file

6. Shutdown the crawler.
   It should close and join the workers if it runs tasks with multiple workers strategy.

Tidying up the description:

What’s the procedure of a crawler working?

PyTsunami - Work Flow



What objects should we have?

1. URL
2. HTTPIO

3. DataHandler
4. PersistenceHandler

What attributes and behaviors should it have?

1. URL
2. HTTPIO
3. DataHandler
4. PersistenceHandler

How many roles does crawler have? What does it do with each roles?

Tidying up above thoughts with UML