NAME

LWP::RobotUA - a class for well-behaved Web robots

SYNOPSIS

```
 use LWP::RobotUA;
 my $ua = LWP::RobotUA->new('my-robot/0.1', 'me@foo.com');
 $ua->delay(10); # be very nice -- max one hit every ten minutes!
 ...

 # Then just use it just like a normal LWP::UserAgent:
  my $response = $ua->get('http://whatever.int/...');
 ...
```

DESCRIPTION

This class implements a user agent that is suitable for robot applications. Robots should be nice to the servers they visit. They should consult the */robots.txt* file to ensure that they are welcomed and they should not make requests too frequently.

But before you consider writing a robot, take a look at <URL:http://www.robotstxt.org/>.

When you use an *LWP::RobotUA* object as your user agent, then you do not really have to think about these things yourself; `robots.txt` files are automatically consulted and obeyed, the server isn't queried too rapidly, and so on. Just send requests as you do when you are using a normal *LWP::UserAgent* object (using `$ua->get(...)`, `$ua->head(...)`, `$ua->request(...)`, etc.), and this special agent will make sure you are nice.

METHODS

The LWP::RobotUA is a sub-class of LWP::UserAgent and implements the same methods. In addition the following methods are provided:

$ua = LWP::RobotUA->new(`%options`)
$ua = LWP::RobotUA->new(`$agent`, `$from`)
$ua = LWP::RobotUA->new(`$agent`, `$from`, `$rules`)

The LWP::UserAgent options `agent` and `from` are mandatory. The options `delay`, `use_sleep` and `rules` initialize attributes private to the RobotUA. If `rules` are not provided, then `WWW::RobotRules` is instantiated providing an internal database of *robots.txt*.

It is also possible to just pass the value of `agent`, `from` and optionally `rules` as plain positional arguments.

$ua->delay
$ua->delay(`$minutes`)

Get/set the minimum delay between requests to the same server, in *minutes*. The default is 1 minute. Note that this number doesn't have to be an integer; for example, this sets the delay to 10 seconds:

```
 $ua->delay(10/60);
```

$ua->use_sleep
$ua->use_sleep(`$boolean`)

Get/set a value indicating whether the UA should *sleep()* if requests arrive too fast, defined as $ua->delay minutes not passed since last request to the given server. The default is TRUE. If this value is FALSE then an internal SERVICE_UNAVAILABLE response will be generated. It will have a Retry-After header that indicates when it is OK to send another request to this server.

$ua->rules
$ua->rules(`$rules`)

Set/get which *WWW::RobotRules* object to use.

`$ua`->no_visits(`$netloc`)
: Returns the number of documents fetched from this server host. Yeah I know, this method should probably have been named *num_visits()* or something like that. :-(

`$ua`->host_wait(`$netloc`)
: Returns the number of *seconds* (from now) you must wait before you can make a new request to this host.

`$ua`->as_string
: Returns a string that describes the state of the UA. Mainly useful for debugging.

SEE ALSO

LWP::UserAgent, WWW::RobotRules

COPYRIGHT