

Addressable

Homepage

github.com/sporkmonger/addressable

Author

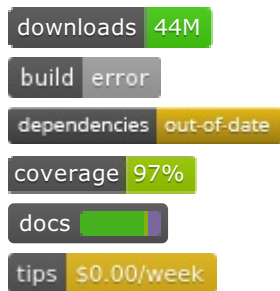
[Bob Aman](#)

Copyright

Copyright © 2006-2016 Bob Aman

License

Apache 2.0



Description

Addressable is a replacement for the URI implementation that is part of Ruby's standard library. It more closely conforms to RFC 3986, RFC 3987, and RFC 6570 (level 4), providing support for IRIs and URI templates.

Reference

- {Addressable::URI}
- {Addressable::Template}

Example usage

```
require "addressable/uri"

uri = Addressable::URI.parse("http://example.com/path/to/resource/")
uri.scheme
#=> "http"
uri.host
#=> "example.com"
uri.path
#=> "/path/to/resource/"

uri = Addressable::URI.parse("http://www.詹姆斯.com/")
uri.normalize
#=> #<Addressable::URI:0xc9a4c8 URI:http://www.xn--8ws00zhy3a.com/>
```

URI Templates

For more details, see [RFC 6570](#).

```
require "addressable/template"

template = Addressable::Template.new("http://example.com/{?query*}/")
template.expand({
  "query" => {
    'foo' => 'bar',
    'color' => 'red'
  }
})
#=> #<Addressable::URI:0xc9d95c URI:http://example.com/?foo=bar&color=red>

template = Addressable::Template.new("http://example.com/{?one,two,three}")
template.partial_expand({"one" => "1", "three" => 3}).pattern
#=> "http://example.com/?one=1{&two}&three=3"

template = Addressable::Template.new(
  "http://{host}/{segments*}/{?one,two,bogus}{#fragment}"
)
uri = Addressable::URI.parse(
  "http://example.com/a/b/c/?one=1&two=2#foo"
)
template.extract(uri)
#=>
# {
#   "host" => "example.com",
#   "segments" => ["a", "b", "c"],
#   "one" => "1",
#   "two" => "2",
#   "fragment" => "foo"
# }
```

Install

```
$ sudo gem install addressable
```

You may optionally turn on native IDN support by installing `libidn` and the `idn` gem:

```
$ sudo apt-get install idn # Debian/Ubuntu
$ brew install libidn # OS X
$ gem install idn-ruby
```