

Ruby - Bug #4677

documentation updates 2011-05-12

05/13/2011 03:20 AM - vbatts (Vincent Batts)

Status:	Closed	
Priority:	Normal	
Assignee:	drbrain (Eric Hodel)	
Target version:	1.9.3	
ruby -v:	Thu May 12 14:17:22 EDT 2011	
Description		Backport:
per http://blog.segment7.net/2011/05/09/ruby-1-9-3-documentation-challenge		
The attached patch addresses the following items needing documentation, including the entirety of URI and subclasses/modules:		
class Regexp		
in file re.c		
IGNORECASE = nil		
in file re.c		
EXTENDED = nil		
in file re.c		
MULTILINE = nil		
in file re.c		
FIXEDENCODING = nil		
in file re.c		
NOENCODING = nil		
end		
module Process # is documented		
in file process.c		
WNOHANG = nil		
in file process.c		
WUNTRACED = nil		
in file process.c		
PRIO_PROCESS = nil		
in file process.c		
PRIO_PGRP = nil		

in file process.c

PRIO_USER = nil

in file process.c

RLIM_SAVED_MAX = nil

in file process.c

RLIM_INFINITY = nil

in file process.c

RLIM_SAVED_CUR = nil

in file process.c

RLIMIT_AS = nil

in file process.c

RLIMIT_CORE = nil

in file process.c

RLIMIT_CPU = nil

in file process.c

RLIMIT_DATA = nil

in file process.c

RLIMIT_FSIZE = nil

in file process.c

RLIMIT_MEMLOCK = nil

in file process.c

RLIMIT_MSGQUEUE = nil

in file process.c

RLIMIT_NICE = nil

in file process.c

RLIMIT_NOFILE = nil

in file process.c

RLIMIT_NPROC = nil

in file process.c

RLIMIT_RSS = nil

in file process.c

RLIMIT_RTPRIO = nil

in file process.c

RLIMIT_RTTIME = nil

in file process.c

RLIMIT_SBSIZE = nil

in file process.c

RLIMIT_SIGPENDING = nil

in file process.c

RLIMIT_STACK = nil

end

module URI # is documented

in file lib/uri/common.rb

DEFAULT_PARSER = nil

in file lib/uri/common.rb

def scheme_list(); end

end

class URI::Generic # is documented

in file lib/uri/generic.rb

DEFAULT_PORT = nil

in file lib/uri/generic.rb

COMPONENT = nil

in file lib/uri/generic.rb

USE_REGISTRY = nil

attr_reader :scheme # in file lib/uri/generic.rb

attr_reader :port # in file lib/uri/generic.rb

attr_reader :registry # in file lib/uri/generic.rb

attr_reader :path # in file lib/uri/generic.rb

attr_reader :query # in file lib/uri/generic.rb

attr_reader :opaque # in file lib/uri/generic.rb

attr_reader :fragment # in file lib/uri/generic.rb

in file lib/uri/generic.rb

def check_fragment(v); end

in file lib/uri/generic.rb

def check_host(v); end

in file lib/uri/generic.rb

def check_opaque(v); end

in file lib/uri/generic.rb

def check_password(v, user = @user); end

in file lib/uri/generic.rb

def check_path(v); end

in file lib/uri/generic.rb

def check_port(v); end

in file lib/uri/generic.rb

def check_query(v); end

in file lib/uri/generic.rb

def check_registry(v); end

in file lib/uri/generic.rb

def check_scheme(v); end

in file lib/uri/generic.rb

def check_user(v); end

in file lib/uri/generic.rb

def check_userinfo(user, password = nil); end

in file lib/uri/generic.rb

def coerce(oth); end

in file lib/uri/generic.rb

def component(); end

in file lib/uri/generic.rb

def component_ary(); end

in file lib/uri/generic.rb

def default_port(); end

in file lib/uri/generic.rb

def escape_userpass(v); end

in file lib/uri/generic.rb

```
def fragment=(v); end
```

in file lib/uri/generic.rb

```
def host=(v); end
```

in file lib/uri/generic.rb

```
def merge_path(base, rel); end
```

in file lib/uri/generic.rb

```
def opaque=(v); end
```

in file lib/uri/generic.rb

```
def parser(); end
```

in file lib/uri/generic.rb

```
def password(); end
```

in file lib/uri/generic.rb

```
def password=(password); end
```

in file lib/uri/generic.rb

```
def path=(v); end
```

in file lib/uri/generic.rb

```
def path_query(); end
```

in file lib/uri/generic.rb

```
def port=(v); end
```

in file lib/uri/generic.rb

```
def query=(v); end
```

in file lib/uri/generic.rb

```
def registry=(v); end
```

in file lib/uri/generic.rb

```
def route_from0(oth); end
```

in file lib/uri/generic.rb

```
def route_from_path(src, dst); end
```

in file lib/uri/generic.rb

def scheme=(v); end

in file lib/uri/generic.rb

def set_fragment(v); end

in file lib/uri/generic.rb

def set_host(v); end

in file lib/uri/generic.rb

def set_opaque(v); end

in file lib/uri/generic.rb

def set_password(v); end

in file lib/uri/generic.rb

def set_path(v); end

in file lib/uri/generic.rb

def set_port(v); end

in file lib/uri/generic.rb

def set_query(v); end

in file lib/uri/generic.rb

def set_registry(v); end

in file lib/uri/generic.rb

def set_scheme(v); end

in file lib/uri/generic.rb

def set_user(v); end

in file lib/uri/generic.rb

def set_userinfo(user, password = nil); end

in file lib/uri/generic.rb

def split_path(path); end

in file lib/uri/generic.rb

def split_userinfo(ui); end

in file lib/uri/generic.rb

def user(); end

in file lib/uri/generic.rb

```
def user=(user); end
```

in file lib/uri/generic.rb

```
def userinfo(); end  
end
```

in files:

lib/uri/common.rb

```
module URI::Escape  
end
```

```
class URI::FTP # is documented
```

in file lib/uri/ftp.rb

```
DEFAULT_PORT = nil
```

in file lib/uri/ftp.rb

```
COMPONENT = nil
```

in file lib/uri/ftp.rb

```
TYPECODE_PREFIX = nil  
attr_reader :typecode # in file lib/uri/ftp.rb
```

in file lib/uri/ftp.rb

```
def new2(user, password, host, port, path, typecode = nil, arg_check = true); end
```

in file lib/uri/ftp.rb

```
def check_typecode(v); end
```

in file lib/uri/ftp.rb

```
def set_path(v); end
```

in file lib/uri/ftp.rb

```
def set_typecode(v); end
```

in file lib/uri/ftp.rb

```
def typecode=(typecode); end  
end
```

```
class URI::HTTP # is documented
```

in file lib/uri/http.rb

```
DEFAULT_PORT = nil
```

in file lib/uri/http.rb

```
COMPONENT = nil  
end
```

```
class URI::HTTPS # is documented
```

in file lib/uri/https.rb

```
DEFAULT_PORT = nil  
end
```

```
class URI::LDAP # is documented
```

in file lib/uri/ldap.rb

```
DEFAULT_PORT = nil
```

in file lib/uri/ldap.rb

```
COMPONENT = nil
```

in file lib/uri/ldap.rb

```
SCOPE = nil
```

in file lib/uri/ldap.rb

```
def build(args); end
```

in file lib/uri/ldap.rb

```
def new(*arg); end
```

in file lib/uri/ldap.rb

```
def attributes(); end
```

in file lib/uri/ldap.rb

```
def attributes=(val); end
```

in file lib/uri/ldap.rb

```
def build_path_query(); end
```

in file lib/uri/ldap.rb

```
def dn(); end
```

in file lib/uri/ldap.rb

```
def dn=(val); end
```

in file lib/uri/ldap.rb

```
def extensions(); end
```

in file lib/uri/ldap.rb

```
def extensions=(val); end
```

in file lib/uri/ldap.rb

```
def filter(); end
```

in file lib/uri/ldap.rb

```
def filter=(val); end
```

in file lib/uri/ldap.rb

```
def parse_dn(); end
```

in file lib/uri/ldap.rb

```
def parse_query(); end
```

in file lib/uri/ldap.rb

```
def scope(); end
```

in file lib/uri/ldap.rb

```
def scope=(val); end
```

in file lib/uri/ldap.rb

```
def set_attributes(val); end
```

in file lib/uri/ldap.rb

```
def set_dn(val); end
```

in file lib/uri/ldap.rb

```
def set_extensions(val); end
```

in file lib/uri/ldap.rb

```
def set_filter(val); end
```

in file lib/uri/ldap.rb

```
def set_scope(val); end
```

```
end
```

```
class URI::LDAPS # is documented
```

in file lib/uri/ldaps.rb

```
DEFAULT_PORT = nil  
end
```

```
class URI::MailTo # is documented
```

in file lib/uri/mailto.rb

```
DEFAULT_PORT = nil
```

in file lib/uri/mailto.rb

```
COMPONENT = nil
```

in file lib/uri/mailto.rb

```
def check_headers(v); end
```

in file lib/uri/mailto.rb

```
def check_to(v); end
```

in file lib/uri/mailto.rb

```
def headers=(v); end
```

in file lib/uri/mailto.rb

```
def set_headers(v); end
```

in file lib/uri/mailto.rb

```
def set_to(v); end
```

in file lib/uri/mailto.rb

```
def to=(v); end
```

```
end
```

in files:

lib/uri/common.rb

```
module URI::REGEXP  
end
```

in files:

lib/uri/common.rb

```
class URI::Parser
```

```
attr_reader :pattern # in file lib/uri/common.rb  
attr_reader :regexp # in file lib/uri/common.rb
```

in file lib/uri/common.rb

```
def escape(str, unsafe = @regexp[:UNSAFE]); end
```

in file lib/uri/common.rb

```
def extract(str, schemes = nil, &block); end
```

in file lib/uri/common.rb

```
def initialize_pattern(opts = {}); end
```

in file lib/uri/common.rb

```
def initialize_regexp(pattern); end
```

in file lib/uri/common.rb

```
def join(*uris); end
```

in file lib/uri/common.rb

```
def make_regexp(schemes = nil); end
```

in file lib/uri/common.rb

```
def parse(uri); end
```

in file lib/uri/common.rb

```
def split(uri); end
```

in file lib/uri/common.rb

```
def unescape(str, escaped = @regexp[:ESCAPED]); end
```

```
end
```

Associated revisions

Revision e2b3183f - 05/12/2011 08:39 PM - drbrain (Eric Hodel)

- re.c (Init_Regexp): Document option constants. Patch by Vincent Batts. [Ruby 1.9 - Bug #4677]
- * lib/uri/common.rb (module URI): Documentation for URI. Patch by Vincent Batts. [Ruby 1.9- Bug #4677]
- * lib/uri/ftp.rb (module URI): ditto
- * lib/uri/generic.rb (module URI): ditto
- * lib/uri/http.rb (module URI): ditto
- * lib/uri/https.rb (module URI): ditto
- * lib/uri/ldap.rb (module URI): ditto
- * lib/uri/ldaps.rb (module URI): ditto
- * lib/uri/mailto.rb (module URI): ditto
- * process.c (Init_process): Document Process constants. Patch by Vincent Batts. [Ruby 1.9- Bug #4677]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@31536 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

History

#1 - 05/13/2011 05:39 AM - drbrain (Eric Hodel)

- Status changed from Open to Closed

- % Done changed from 0 to 100

This issue was solved with changeset r31536.

Vincent, thank you for reporting this issue.

Your contribution to Ruby is greatly appreciated.

May Ruby be with you.

-
- re.c (Init_Regexp): Document option constants. Patch by Vincent Batts. [Ruby 1.9 - Bug [#4677](#)]

- lib/uri/common.rb (module URI): Documentation for URI. Patch by Vincent Batts. [Ruby 1.9- Bug [#4677](#)]
- lib/uri/ftp.rb (module URI): ditto
- lib/uri/generic.rb (module URI): ditto
- lib/uri/http.rb (module URI): ditto
- lib/uri/https.rb (module URI): ditto
- lib/uri/ldap.rb (module URI): ditto
- lib/uri/ldaps.rb (module URI): ditto
- lib/uri/mailto.rb (module URI): ditto
- process.c (Init_process): Document Process constants. Patch by Vincent Batts. [Ruby 1.9- Bug [#4677](#)]

#2 - 05/13/2011 05:39 AM - drbrain (Eric Hodel)

For the re.c patch I moved the documentation above the constant definition. RDoc doesn't pick up documentation on the same line for C files.

Files

ruby_doc_updates-20110512_vbatts.patch	39 KB	05/13/2011	vbatts (Vincent Batts)
--	-------	------------	------------------------