

NAME

XML::LibXML::Schema – XML Schema Validation

SYNOPSIS

```
use XML::LibXML;
$doc = XML::LibXML->new->parse_file($url);

$xmlschema = XML::LibXML::Schema->new( location => $filename_or_url );
$xmlschema = XML::LibXML::Schema->new( string => $xmlschemastring );
eval { $xmlschema->validate( $doc ); };
```

DESCRIPTION

The XML::LibXML::Schema class is a tiny frontend to libxml2's XML Schema implementation. Currently it supports only schema parsing and document validation. As of 2.6.32, libxml2 only supports decimal types up to 24 digits (the standard requires at least 18).

METHODS

new

```
$xmlschema = XML::LibXML::Schema->new( location => $filename_or_url );
$xmlschema = XML::LibXML::Schema->new( string => $xmlschemastring );
```

The constructor of XML::LibXML::Schema may get called with either one of two parameters. The parameter tells the class from which source it should generate a validation schema. It is important, that each schema only have a single source.

The location parameter allows one to parse a schema from the filesystem or a URL.

The string parameter will parse the schema from the given XML string.

Note that the constructor will **die()** if the schema does not meet the constraints of the XML Schema specification.

validate

```
eval { $xmlschema->validate( $doc ); };
```

This function allows one to validate a (parsed) document against the given XML Schema. The argument of this function should be a XML::LibXML::Document object. If this function succeeds, it will return 0, otherwise it will **die()** and report the errors found. Because of this **validate()** should be always evaluated.

AUTHORS

Matt Sergeant, Christian Glahn, Petr Pajas

VERSION

2.0134

COPYRIGHT

2001–2007, AxKit.com Ltd.

2002–2006, Christian Glahn.

2006–2009, Petr Pajas.

LICENSE

This program is free software; you can redistribute it and/or modify it under the same terms as Perl itself.