SFML

* [Main Page](http://docs.google.com/index.htm)
* [Modules](http://docs.google.com/modules.htm)
* [Classes](http://docs.google.com/annotated.htm)
* [Files](http://docs.google.com/files.htm)
* [File List](http://docs.google.com/files.htm)
* [include](http://docs.google.com/dir_f3190241575fd2bd132a392ae6942f4a.htm)
* [SFML](http://docs.google.com/dir_692f376662c82a26cfe4cfa3aceebe24.htm)
* [Network](http://docs.google.com/dir_b9ac88db2949395b3130dd4ffb1be4e1.htm)

IpAddress.hpp

1

2 //

3 // SFML - Simple and Fast Multimedia Library

4 // Copyright (C) 2007-2013 Laurent Gomila (laurent.gom@gmail.com)

5 //

6 // This software is provided 'as-is', without any express or implied warranty.

7 // In no event will the authors be held liable for any damages arising from the use of this software.

8 //

9 // Permission is granted to anyone to use this software for any purpose,

10 // including commercial applications, and to alter it and redistribute it freely,

11 // subject to the following restrictions:

12 //

13 // 1. The origin of this software must not be misrepresented;

14 // you must not claim that you wrote the original software.

15 // If you use this software in a product, an acknowledgment

16 // in the product documentation would be appreciated but is not required.

17 //

18 // 2. Altered source versions must be plainly marked as such,

19 // and must not be misrepresented as being the original software.

20 //

21 // 3. This notice may not be removed or altered from any source distribution.

22 //

24

25 #ifndef SFML\_IPADDRESS\_HPP

26 #define SFML\_IPADDRESS\_HPP

27

29 // Headers

31 #include <SFML/Network/Export.hpp>

32 #include <SFML/System/Time.hpp>

33 #include <istream>

34 #include <ostream>

35 #include <string>

36

37

38 namespace sf

39 {

[44](http://docs.google.com/classsf_1_1IpAddress.htm) class SFML\_NETWORK\_API [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)

45 {

46 public :

47

54  [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)();

55

65  [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)(const std::string& address);

66

79  [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)(const char\* address);

80

94  [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)(Uint8 byte0, Uint8 byte1, Uint8 byte2, Uint8 byte3);

95

109  explicit [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)(Uint32 address);

110

123  std::string toString() const;

124

139  Uint32 toInteger() const;

140

155  static [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm) getLocalAddress();

156

179  static [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm) getPublicAddress([Time](http://docs.google.com/classsf_1_1Time.htm) timeout = [Time::Zero](http://docs.google.com/classsf_1_1Time.htm#a8db127b632fa8da21550e7282af11fa0));

180

182  // Static member data

[184](http://docs.google.com/classsf_1_1IpAddress.htm#a4619b4abbe3c8fef056e7299db967404)  static const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm) [None](http://docs.google.com/classsf_1_1IpAddress.htm#a4619b4abbe3c8fef056e7299db967404);

[185](http://docs.google.com/classsf_1_1IpAddress.htm#a594d3a8e2559f8fa8ab0a96fa597333b)  static const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm) [LocalHost](http://docs.google.com/classsf_1_1IpAddress.htm#a594d3a8e2559f8fa8ab0a96fa597333b);

[186](http://docs.google.com/classsf_1_1IpAddress.htm#aa93d1d57b65d243f2baf804b6035465c)  static const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm) [Broadcast](http://docs.google.com/classsf_1_1IpAddress.htm#aa93d1d57b65d243f2baf804b6035465c);

187

188 private :

189

191  // Member data

193  Uint32 m\_address;

194 };

195

205 SFML\_NETWORK\_API bool operator ==(const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& left, const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& right);

206

216 SFML\_NETWORK\_API bool operator !=(const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& left, const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& right);

217

227 SFML\_NETWORK\_API bool operator <(const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& left, const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& right);

228

238 SFML\_NETWORK\_API bool operator >(const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& left, const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& right);

239

249 SFML\_NETWORK\_API bool operator <=(const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& left, const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& right);

250

260 SFML\_NETWORK\_API bool operator >=(const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& left, const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& right);

261

271 SFML\_NETWORK\_API std::istream& operator >>(std::istream& stream, [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& address);

272

282 SFML\_NETWORK\_API std::ostream& operator <<(std::ostream& stream, const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& address);

283

284 } // namespace sf

285

286

287 #endif // SFML\_IPADDRESS\_HPP

288

289

Copyright � Laurent Gomila  ::  Documentation generated by [doxygen](http://www.doxygen.org/)  ::