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)

UdpSocket.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\_UDPSOCKET\_HPP

26 #define SFML\_UDPSOCKET\_HPP

27

29 // Headers

31 #include <SFML/Network/Export.hpp>

32 #include <SFML/Network/Socket.hpp>

33 #include <vector>

34

35

36 namespace sf

37 {

38 class IpAddress;

39 class Packet;

40

[45](http://docs.google.com/classsf_1_1UdpSocket.htm) class SFML\_NETWORK\_API [UdpSocket](http://docs.google.com/classsf_1_1UdpSocket.htm) : public [Socket](http://docs.google.com/classsf_1_1Socket.htm)

46 {

47 public :

48

50  // Constants

52  enum

53  {

[54](http://docs.google.com/classsf_1_1UdpSocket.htm#a14c7b7816e33ed1ef1f2fdb2404c06b5a728a7d33027bee0d65f70f964dd9c9eb)  MaxDatagramSize = 65507

55  };

56

61  [UdpSocket](http://docs.google.com/classsf_1_1UdpSocket.htm)();

62

74  unsigned short getLocalPort() const;

75

92  Status bind(unsigned short port);

93

104  void unbind();

105

123  Status send(const void\* data, std::size\_t size, const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& remoteAddress, unsigned short remotePort);

124

146  Status receive(void\* data, std::size\_t size, std::size\_t& received, [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& remoteAddress, unsigned short& remotePort);

147

164  Status send([Packet](http://docs.google.com/classsf_1_1Packet.htm)& packet, const [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& remoteAddress, unsigned short remotePort);

165

181  Status receive([Packet](http://docs.google.com/classsf_1_1Packet.htm)& packet, [IpAddress](http://docs.google.com/classsf_1_1IpAddress.htm)& remoteAddress, unsigned short& remotePort);

182

183 private:

184

186  // Member data

188  std::vector<char> m\_buffer;

189 };

190

191 } // namespace sf

192

193

194 #endif // SFML\_UDPSOCKET\_HPP

195

196

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