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)
* [System](http://docs.google.com/dir_60c5c649f8df3b69a45a020d59f81335.htm)

ThreadLocalPtr.inl

1

2 //

3 // SFML - Simple and Fast Multimedia Library

4 // Copyright (C) 2007-2012 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

26 namespace sf

27 {

29 template <typename T>

[30](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm#a8c678211d7828d2a8c41cb534422d649) [ThreadLocalPtr<T>::ThreadLocalPtr](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm#a8c678211d7828d2a8c41cb534422d649)(T\* value) :

31 [ThreadLocal](http://docs.google.com/classsf_1_1ThreadLocal.htm)(value)

32 {

33 }

34

35

37 template <typename T>

[38](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm#aa3bac9a08e8739613961659d10e0fadd) T& [ThreadLocalPtr<T>::operator \*](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm#aa3bac9a08e8739613961659d10e0fadd)() const

39 {

40  return \*static\_cast<T\*>(getValue());

41 }

42

43

45 template <typename T>

[46](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm#aa0b559f78929b22cb2585cb2966edfb2) T\* [ThreadLocalPtr<T>::operator ->](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm#aa0b559f78929b22cb2585cb2966edfb2)() const

47 {

48  return static\_cast<T\*>(getValue());

49 }

50

51

53 template <typename T>

[54](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm#ab4a6a341c26b58f0ed3ef86502bd9572) [ThreadLocalPtr<T>::operator](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm) T\*() const

55 {

56  return static\_cast<T\*>(getValue());

57 }

58

59

61 template <typename T>

[62](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm#a14dcf1cdf5f6b3bcdd633014b2b671f5) [ThreadLocalPtr<T>](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm)& [ThreadLocalPtr<T>::operator =](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm#a14dcf1cdf5f6b3bcdd633014b2b671f5)(T\* value)

63 {

64  setValue(value);

65  return \*this;

66 }

67

68

70 template <typename T>

[71](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm#a6792a6a808af06f0d13e3ceecf2fc947) [ThreadLocalPtr<T>](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm)& [ThreadLocalPtr<T>::operator =](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm#a14dcf1cdf5f6b3bcdd633014b2b671f5)(const [ThreadLocalPtr<T>](http://docs.google.com/classsf_1_1ThreadLocalPtr.htm)& right)

72 {

73  setValue(right.[getValue](http://docs.google.com/classsf_1_1ThreadLocal.htm#aef35a39686eac4b6634a6e5605aacfd7)());

74  return \*this;

75 }

76

77 } // namespace sf

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