| LCOV - code coverage report |
| --- |
|  |
| | Current view: | [top level](http://docs.google.com/index.html) - [Output/include](http://docs.google.com/index.html) - OutputBuffer.h (source / [functions](http://docs.google.com/OutputBuffer.h.func.html)) |  |  | Hit | Total | Coverage | | --- | --- | --- | --- | --- | --- | --- | | Test: | EBM9K Converter Coverage Report |  | Lines: | 1 | 1 | 100.0 % | | Date: | 2012-03-23 |  | Functions: | 0 | 0 | - | | Legend: | Lines: hit not hit | Branches: + taken - not taken # not executed |  | Branches: | 0 | 0 | - | |  | |
|  |

|  |
| --- |
| Branch data Line data Source code  1 : : #ifndef \_OUTPUTBUFFER\_H  2 : : #define \_OUTPUTBUFFER\_H  3 : :   4 : : #include <cstddef>  5 : :   6 : : #include "DPL.h"  7 : :   8 : : #include "MemInfo.h"  9 : : #include "Memory.h"  10 : :   11 : : namespace nft {  12 : :   13 : : class OutputBuffer {  14 : : public:  15 : : typedef OutputBuffer \* Pointer;  16 : : static const size\_t cIncrement;  17 : :   18 : : public:  19 : :   20 : : OutputBuffer(size\_t inInitialSize);  21 : :   22 : : ~OutputBuffer();  23 : :   24 : : OutputBuffer(const OutputBuffer & source);  25 : :   26 : : OutputBuffer &  27 : : operator=(const OutputBuffer & source);  28 : :   29 : : void \*  30 : : getWritableMemory(size\_t inSize) throw(DPL::Error \*);  31 : :   32 : : MemInfo  33 : : getOutputData() const;  34 : :   35 : : inline char \*  36 : : getBuffer() throw();  37 : :   38 : : void  39 : : reset();  40 : :   41 : : private:  42 : : size\_t mMemorySize;  43 : : size\_t mCurrentOffset;  44 : : char \* mBuffer;  45 : : };  46 : :   47 : : inline char \*  48 : 3196 : OutputBuffer::getBuffer() throw() {  49 : : return mBuffer;  50 : : }  51 : :   52 : : } //namesapce nft  53 : :   54 : : #endif |

|  |
| --- |
| Generated by: [LCOV version 1.9](http://ltp.sourceforge.net/coverage/lcov.php) |