| LCOV - code coverage report |
| --- |
|  |
| | Current view: | [top level](http://docs.google.com/index.html) - [TimeCalculators/src](http://docs.google.com/index.html) - TimeData.cpp (source / [functions](http://docs.google.com/TimeData.cpp.func.html)) |  |  | Hit | Total | Coverage | | --- | --- | --- | --- | --- | --- | --- | | Test: | EBM9K Converter Coverage Report |  | Lines: | 37 | 50 | 74.0 % | | Date: | 2012-03-23 |  | Functions: | 6 | 9 | 66.7 % | | Legend: | Lines: hit not hit | Branches: + taken - not taken # not executed |  | Branches: | 11 | 20 | 55.0 % | |  | |
|  |

|  |
| --- |
| Branch data Line data Source code  1 : : #include "TimeData.h"  2 : : #include "ErrorHandler.h"  3 : : #include "Logger.h"  4 : : #include "ConverterParams.h"  5 : : #include "print.h"  6 : :   7 : : namespace nft {  8 : :   9 : : const uint16\_t TimeData::MARGIN1 = 5;  10 : : const uint16\_t TimeData::MARGIN2 = 10;  11 : :   12 : : TimeData::TimeData(  13 : : uint64\_t numColumns, int64\_t originX, int64\_t pitch)  14 : 1092 : :mNumColumns( numColumns ),  15 : : mOriginX( originX ),  16 : : mPitch( pitch )  17 : 1092 : {  18 : 1092 : mOriginX -= mPitch \* TimeData::MARGIN1;  19 : 1092 : mNumColumns += TimeData::MARGIN1 \* 2;  20 : :   21 : 1092 : mTimeDataSize = sizeof( DPMANAGER\_MESSAGE\_SHOT\_TIMEDATA )  22 : : \* mNumColumns;  23 : :   24 : 1092 : mTimeData = new DPMANAGER\_MESSAGE\_SHOT\_TIMEDATA[mNumColumns];  25 : 1092 : bzero( mTimeData, mTimeDataSize );  26 : : }  27 : :   28 : : TimeData::~TimeData()  29 : 2184 : {  30 [ # # ][ + - ]: 1092 : if(NULL != mTimeData){  31 : 1092 : delete[] mTimeData;  32 : 1092 : mTimeData = NULL;  33 : : }  34 : : //Do not call delete on mTimeData  35 : : }  36 : :   37 : : TimeData::TimeData(const TimeData & source)  38 : 0 : :mTimeData( source.mTimeData ),  39 : : mNumColumns( source.mNumColumns ),  40 : : mOriginX( source.mOriginX ),  41 : : mPitch( source.mPitch )  42 : 0 : {  43 : :   44 : : }  45 : :   46 : : TimeData &  47 : : TimeData::operator=(const TimeData & source)  48 : 0 : {  49 [ # # ]: 0 : if( this != &source )  50 : : {  51 : 0 : mTimeData = source.mTimeData;  52 : 0 : mNumColumns = source.mNumColumns;  53 : 0 : mOriginX = source.mOriginX;  54 : 0 : mPitch = source.mPitch;  55 : : }  56 : :   57 : 0 : return \*this;  58 : : }  59 : :   60 : : void  61 : : TimeData::updateDrawingTimes(  62 : : int64\_t sfPosX, uint64\_t time) throw(DPL::Error \*)  63 : 2179 : {  64 : 2179 : int64\_t index, margin;  65 : 2179 : size\_t lastDataSize;  66 : :   67 : 2179 : index = (sfPosX - mOriginX) / mPitch;  68 : :   69 : 2179 : e9k\_debug("Time data index = %lld\n", index );  70 : : //#ifdef ERROR\_SEED //<<Test\_Id:tmp\_pTimeData\_Set\_To\_NULL>>  71 : : // index = -1;  72 : : //#endif   73 [ + + ][ + + ]: 2179 : if( index < 0 || index >= static\_cast<int64\_t>(mNumColumns) ){  74 : :   75 : 11 : margin = TimeData::MARGIN2  76 : : + ((index < 0) ? -index : (index - mNumColumns));  77 : :   78 : 11 : mNumColumns += margin \* 2;  79 : 11 : lastDataSize = mTimeDataSize;  80 : 11 : mTimeDataSize = sizeof( DPMANAGER\_MESSAGE\_SHOT\_TIMEDATA )  81 : : \* mNumColumns;  82 : :   83 : 11 : DPMANAGER\_MESSAGE\_SHOT\_TIMEDATA \* tmp\_pTimeData;  84 : 11 : tmp\_pTimeData = new(std::nothrow)  85 : : DPMANAGER\_MESSAGE\_SHOT\_TIMEDATA[mNumColumns];  86 : : //#ifdef ERROR\_SEED //<<Test\_Id:tmp\_pTimeData\_Set\_To\_NULL>>  87 : : // tmp\_pTimeData = NULL;  88 : : //#endif  89 [ - + ]: 11 : if(NULL == tmp\_pTimeData)  90 : : {  91 : 0 : int iErrNo = errno; //System error no  92 : 0 : char systemErrorMsg[BUFSIZ];  93 : 0 : ErrorHandler \*pErr = ErrorHandler::instance();  94 : 0 : pErr->errorAdd(NULL, \_\_FILE\_\_, \_\_FUNCTION\_\_, \_\_LINE\_\_,  95 : : ErrorHandler::eErrMemoryAllocation,  96 : : pErr->getErrorMsg(  97 : : ErrorHandler::eErrMemoryAllocation,  98 : : strerror\_r( iErrNo, systemErrorMsg, BUFSIZ )));  99 : : }  100 : 11 : bzero( tmp\_pTimeData, mTimeDataSize);  101 : :   102 [ + + ]: 11 : if( index < 0 ) {  103 : 5 : mOriginX -= mPitch \* margin \* 2;  104 : 5 : index = ( sfPosX - mOriginX ) / mPitch;  105 : 5 : memcpy( tmp\_pTimeData + margin\*2, mTimeData, lastDataSize );  106 : : }  107 : : else  108 : : {  109 : 6 : memcpy( tmp\_pTimeData, mTimeData, lastDataSize );  110 : : }  111 [ + - ]: 11 : delete [] mTimeData;  112 : 2179 : mTimeData = tmp\_pTimeData;  113 : : }  114 : :   115 : 2179 : mTimeData[ index ].ullSFTotalTime += time;  116 [ + + ]: 2179 : mTimeData[ index ].ullSFMaxTime   117 : : = ( time > mTimeData[ index ].ullSFMaxTime)   118 : : ? time : mTimeData[ index ].ullSFMaxTime;  119 [ # # ]: 2179 : return;  120 : : }  121 : :   122 : :   123 : 1215 : } // namespace nft |

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