#pragma once

#ifdef \_\_cplusplus

#include <twml/defines.h>

#include <twml/TensorRecord.h>

namespace twml {

// Encodes tensors as DataRecord/TensorRecord-compatible Thrift.

// DataRecordWriter relies on this class to encode the tensor fields.

class TWMLAPI TensorRecordWriter {

private:

uint32\_t m\_records\_written;

twml::ThriftWriter &m\_thrift\_writer;

void writeTensor(const RawTensor &tensor);

void writeRawTensor(const RawTensor &tensor);

public:

TensorRecordWriter(twml::ThriftWriter &thrift\_writer):

m\_records\_written(0),

m\_thrift\_writer(thrift\_writer) { }

uint32\_t getRecordsWritten();

// Caller (usually DataRecordWriter) must precede with struct header field

// like thrift\_writer.writeStructFieldHeader(TTYPE\_MAP, DR\_GENERAL\_TENSOR)

//

// All tensors written as RawTensors except for StringTensors

uint64\_t write(twml::TensorRecord &record);

};

}

#endif