#pragma once

#ifdef \_\_cplusplus

#include <twml/defines.h>

#include <twml/RawTensor.h>

#include <cstdint>

#include <unordered\_map>

namespace twml {

class TensorRecordReader;

// A class containing the data from TensorRecord.

// - This serves as the base class from which DataRecord and HashedDataRecord are inherited.

class TWMLAPI TensorRecord {

public:

typedef std::unordered\_map<int64\_t, const RawTensor> RawTensors;

typedef std::unordered\_map<int64\_t, const RawSparseTensor> RawSparseTensors;

private:

RawTensors m\_tensors;

RawSparseTensors m\_sparse\_tensors;

public:

const RawTensors &getRawTensors() {

return m\_tensors;

}

const RawTensor& getRawTensor(int64\_t id) const {

return m\_tensors.at(id);

}

const RawSparseTensor& getRawSparseTensor(int64\_t id) const {

return m\_sparse\_tensors.at(id);

}

void addRawTensor(int64\_t id, const RawTensor &tensor) {

m\_tensors.emplace(id, tensor);

}

friend class TensorRecordReader;

};

}

#endif