package com.twitter.search.core.earlybird.index.column;

import java.io.IOException;

import org.apache.lucene.index.LeafReader;

import org.apache.lucene.index.NumericDocValues;

import com.twitter.search.core.earlybird.index.DocIDToTweetIDMapper;

/\*\*

\* Get an underlying data for a field by calling

\* EarlybirdIndexSegmentAtomicReader#getNumericDocValues(String).

\*/

public abstract class ColumnStrideFieldIndex {

private final String name;

public ColumnStrideFieldIndex(String name) {

this.name = name;

}

public String getName() {

return name;

}

/\*\*

\* Returns the CSF value for the given doc ID.

\*/

public abstract long get(int docID);

/\*\*

\* Updates the CSF value for the given doc ID to the given value.

\*/

public void setValue(int docID, long value) {

throw new UnsupportedOperationException();

}

/\*\*

\* Loads the CSF from an AtomicReader.

\*/

public void load(LeafReader atomicReader, String field) throws IOException {

NumericDocValues docValues = atomicReader.getNumericDocValues(field);

if (docValues != null) {

for (int i = 0; i < atomicReader.maxDoc(); i++) {

if (docValues.advanceExact(i)) {

setValue(i, docValues.longValue());

}

}

}

}

/\*\*

\* Optimizes the representation of this column stride field, and remaps its doc IDs, if necessary.

\*

\* @param originalTweetIdMapper The original tweet ID mapper.

\* @param optimizedTweetIdMapper The optimized tweet ID mapper.

\* @return An optimized column stride field equivalent to this CSF,

\* with possibly remapped doc IDs.

\*/

public ColumnStrideFieldIndex optimize(

DocIDToTweetIDMapper originalTweetIdMapper,

DocIDToTweetIDMapper optimizedTweetIdMapper) throws IOException {

return this;

}

}