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

import java.util.HashMap;

import java.util.Iterator;

import java.util.Map;

import org.apache.lucene.index.Fields;

import org.apache.lucene.index.Terms;

public class InMemoryFields extends Fields {

private final Map<InvertedIndex, Terms> termsCache = new HashMap<>();

private final Map<String, InvertedIndex> perFields;

private final Map<InvertedIndex, Integer> pointerIndex;

/\*\*

\* Returns a new {@link Fields} instance for the provided {@link InvertedIndex}es.

\*/

public InMemoryFields(Map<String, InvertedIndex> perFields,

Map<InvertedIndex, Integer> pointerIndex) {

this.perFields = perFields;

this.pointerIndex = pointerIndex;

}

@Override

public Iterator<String> iterator() {

return perFields.keySet().iterator();

}

@Override

public Terms terms(String field) {

InvertedIndex invertedIndex = perFields.get(field);

if (invertedIndex == null) {

return null;

}

return termsCache.computeIfAbsent(invertedIndex,

index -> index.createTerms(pointerIndex.getOrDefault(invertedIndex, -1)));

}

@Override

public int size() {

return perFields.size();

}

}