package com.twitter.search.common.search.termination;

import java.io.IOException;

import com.google.common.base.Preconditions;

import org.apache.lucene.search.DocIdSetIterator;

import org.apache.lucene.search.Scorer;

import org.apache.lucene.search.Weight;

import com.twitter.search.common.metrics.SearchRateCounter;

import com.twitter.search.common.query.FilteredScorer;

import com.twitter.search.common.search.DocIdTracker;

/\*\*

\* Scorer implementation that adds termination support for an underlying query.

\* Meant to be used in conjunction with {@link TerminationQuery}.

\*/

public class TerminationQueryScorer extends FilteredScorer implements DocIdTracker {

private final QueryTimeout timeout;

private int lastSearchedDocId = -1;

TerminationQueryScorer(Weight weight, Scorer inner, QueryTimeout timeout) {

super(weight, inner);

this.timeout = Preconditions.checkNotNull(timeout);

this.timeout.registerDocIdTracker(this);

SearchRateCounter.export(

timeout.getClientId() + "\_num\_termination\_query\_scorers\_created").increment();

}

@Override

public DocIdSetIterator iterator() {

final DocIdSetIterator superDISI = super.iterator();

return new DocIdSetIterator() {

// lastSearchedDocId is the ID of the last document that was traversed in the posting list.

// docId is the current doc ID in this iterator. In most cases, lastSearchedDocId and docId

// will be equal. They will be different only if the query needed to be terminated based on

// the timeout. In that case, docId will be set to NO\_MORE\_DOCS, but lastSearchedDocId will

// still be set to the last document that was actually traversed.

private int docId = -1;

@Override

public int docID() {

return docId;

}

@Override

public int nextDoc() throws IOException {

if (docId == NO\_MORE\_DOCS) {

return NO\_MORE\_DOCS;

}

if (timeout.shouldExit()) {

docId = NO\_MORE\_DOCS;

} else {

docId = superDISI.nextDoc();

lastSearchedDocId = docId;

}

return docId;

}

@Override

public int advance(int target) throws IOException {

if (docId == NO\_MORE\_DOCS) {

return NO\_MORE\_DOCS;

}

if (target == NO\_MORE\_DOCS) {

docId = NO\_MORE\_DOCS;

lastSearchedDocId = docId;

} else if (timeout.shouldExit()) {

docId = NO\_MORE\_DOCS;

} else {

docId = superDISI.advance(target);

lastSearchedDocId = docId;

}

return docId;

}

@Override

public long cost() {

return superDISI.cost();

}

};

}

@Override

public int getCurrentDocId() {

return lastSearchedDocId;

}

}