package com.twitter.search.earlybird.search.facets;

import java.io.IOException;

import org.apache.lucene.index.NumericDocValues;

import com.twitter.search.common.schema.base.Schema;

import com.twitter.search.common.schema.earlybird.EarlybirdFieldConstants.EarlybirdFieldConstant;

import com.twitter.search.core.earlybird.facets.CSFFacetCountIterator;

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

/\*\*

\* And iterator for counting retweets. Reads from shared\_status\_id CSF but doesn't count

\* replies.

\*/

public class RetweetFacetCountIterator extends CSFFacetCountIterator {

private final NumericDocValues featureReaderIsRetweetFlag;

public RetweetFacetCountIterator(

EarlybirdIndexSegmentAtomicReader reader,

Schema.FieldInfo facetFieldInfo) throws IOException {

super(reader, facetFieldInfo);

featureReaderIsRetweetFlag =

reader.getNumericDocValues(EarlybirdFieldConstant.IS\_RETWEET\_FLAG.getFieldName());

}

@Override

protected boolean shouldCollect(int internalDocID, long termID) throws IOException {

// termID == 0 means that we didn't set shared\_status\_csf, so don't collect

// (tweet IDs are all positive)

// Also only collect if this doc is a retweet, not a reply

return termID > 0

&& featureReaderIsRetweetFlag.advanceExact(internalDocID)

&& (featureReaderIsRetweetFlag.longValue() != 0);

}

}