package com.twitter.search.earlybird.search;

import java.io.IOException;

import java.util.Map;

import org.apache.lucene.index.IndexReader;

import org.apache.lucene.index.Term;

import org.apache.lucene.search.IndexSearcher;

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

import com.twitter.search.earlybird.EarlybirdSearcher;

import com.twitter.search.earlybird.search.facets.AbstractFacetTermCollector;

import com.twitter.search.earlybird.search.facets.FacetResultsCollector;

import com.twitter.search.earlybird.search.facets.TermStatisticsCollector.TermStatisticsSearchResults;

import com.twitter.search.earlybird.search.facets.TermStatisticsRequestInfo;

import com.twitter.search.earlybird.thrift.ThriftFacetCount;

import com.twitter.search.earlybird.thrift.ThriftFacetFieldResults;

import com.twitter.search.earlybird.thrift.ThriftSearchResults;

import com.twitter.search.earlybird.thrift.ThriftTermStatisticsResults;

public abstract class EarlybirdLuceneSearcher extends IndexSearcher {

public EarlybirdLuceneSearcher(IndexReader r) {

super(r);

}

/\*\*

\* Fills facet information for all given search results.

\*

\* @param collector A collector that knows how collect facet information.

\* @param searchResults The search results.

\*/

public abstract void fillFacetResults(

AbstractFacetTermCollector collector, ThriftSearchResults searchResults)

throws IOException;

/\*\*

\* Fills metadata for all given facet results.

\*

\* @param facetResults The facet results.

\* @param schema The earlybird schema.

\* @param debugMode The debug mode for the request that yielded these results.

\*/

public abstract void fillFacetResultMetadata(

Map<Term, ThriftFacetCount> facetResults,

ImmutableSchemaInterface schema,

byte debugMode) throws IOException;

/\*\*

\* Fills metadata for all given term stats results.

\*

\* @param termStatsResults The term stats results.

\* @param schema The earlybird schema.

\* @param debugMode The debug mode for the request that yielded these results.

\*/

public abstract void fillTermStatsMetadata(

ThriftTermStatisticsResults termStatsResults,

ImmutableSchemaInterface schema,

byte debugMode) throws IOException;

/\*\*

\* Returns the results for the given term stats request.

\*

\* @param searchRequestInfo Stores the original term stats request and some other useful request

\* information.

\* @param searcher The searcher that should be used to execute the request.

\* @param requestDebugMode The debug mode for this request.

\* @return The term stats results for the given request.

\*/

public abstract TermStatisticsSearchResults collectTermStatistics(

TermStatisticsRequestInfo searchRequestInfo,

EarlybirdSearcher searcher,

int requestDebugMode) throws IOException;

/\*\*

\* Writes an explanation for the given hits into the given ThriftSearchResults instance.

\*

\* @param searchRequestInfo Stores the original request and some other useful request context.

\* @param hits The hits.

\* @param searchResults The ThriftSearchResults where the explanation for the given hits will be

\* stored.

\*/

// Writes explanations into the searchResults thrift.

public abstract void explainSearchResults(SearchRequestInfo searchRequestInfo,

SimpleSearchResults hits,

ThriftSearchResults searchResults) throws IOException;

public static class FacetSearchResults extends SearchResultsInfo {

private FacetResultsCollector collector;

public FacetSearchResults(FacetResultsCollector collector) {

this.collector = collector;

}

public ThriftFacetFieldResults getFacetResults(String facetName, int topK) {

return collector.getFacetResults(facetName, topK);

}

}

}