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

import java.util.Map;

import java.util.Set;

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

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

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

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

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

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

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

public abstract class AbstractFacetTermCollector implements FacetTermCollector {

private Map<String, FacetLabelProvider> facetLabelProviders;

private FacetIDMap facetIdMap;

/\*\*

\* Populates the given ThriftSearchResult instance with the results collected by this collector

\* and clears all collected results in this collector.

\*

\* @param result The ThriftSearchResult instance to be populated with the results collected in

\* this collector.

\*/

public abstract void fillResultAndClear(ThriftSearchResult result);

public void resetFacetLabelProviders(

Map<String, FacetLabelProvider> facetLabelProvidersToReset, FacetIDMap facetIdMapToReset) {

this.facetLabelProviders = facetLabelProvidersToReset;

this.facetIdMap = facetIdMapToReset;

}

String findFacetName(int fieldId) {

return fieldId < 0 ? null : facetIdMap.getFacetFieldByFacetID(fieldId).getFacetName();

}

protected ThriftSearchResultExtraMetadata getExtraMetadata(ThriftSearchResult result) {

ThriftSearchResultMetadata metadata = result.getMetadata();

if (!metadata.isSetExtraMetadata()) {

metadata.setExtraMetadata(new ThriftSearchResultExtraMetadata());

}

return metadata.getExtraMetadata();

}

protected String getTermFromProvider(

String facetName, long termID, FacetLabelProvider provider) {

return provider.getLabelAccessor().getTermText(termID);

}

protected String getTermFromFacet(long termID, int fieldID, Set<String> facetsToCollectFrom) {

if (termID == EarlybirdIndexSegmentAtomicReader.TERM\_NOT\_FOUND) {

return null;

}

String facetName = findFacetName(fieldID);

if (!facetsToCollectFrom.contains(facetName)) {

return null;

}

final FacetLabelProvider provider = facetLabelProviders.get(facetName);

if (provider == null) {

return null;

}

return getTermFromProvider(facetName, termID, provider);

}

}