package com.twitter.search.earlybird\_root.caching;

import com.twitter.search.common.caching.Cache;

import com.twitter.search.common.caching.CacheUtil;

import com.twitter.search.common.caching.filter.ServicePostProcessor;

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

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

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

import com.twitter.search.earlybird\_root.common.EarlybirdRequestContext;

public class RelevanceZeroResultsServicePostProcessor

extends ServicePostProcessor<EarlybirdRequestContext, EarlybirdResponse> {

private static final SearchCounter RELEVANCE\_RESPONSES\_WITH\_ZERO\_RESULTS\_COUNTER =

SearchCounter.export("relevance\_responses\_with\_zero\_results");

private final Cache<EarlybirdRequest, EarlybirdResponse> cache;

public RelevanceZeroResultsServicePostProcessor(

Cache<EarlybirdRequest, EarlybirdResponse> cache) {

this.cache = cache;

}

@Override

public void processServiceResponse(EarlybirdRequestContext requestContext,

EarlybirdResponse serviceResponse) {

// serviceResponse is the response to a personalized query. If it has zero results, then we can

// cache it and reuse it for other requests with the same query. Otherwise, it makes no sense to

// cache this response.

if (!CacheCommonUtil.hasResults(serviceResponse)) {

RELEVANCE\_RESPONSES\_WITH\_ZERO\_RESULTS\_COUNTER.increment();

CacheUtil.cacheResults(

cache, requestContext.getRequest(), serviceResponse, Integer.MAX\_VALUE);

}

}

}