package com.twitter.search.earlybird.partition;

import com.google.common.base.Predicate;

import com.twitter.search.common.util.hash.EarlybirdPartitioningFunction;

import com.twitter.search.common.util.hash.GeneralEarlybirdPartitioningFunction;

public final class UserPartitionUtil {

private UserPartitionUtil() {

}

/\*\*

\* Filter out the users that are not present in this partition.

\*/

public static Predicate<Long> filterUsersByPartitionPredicate(final PartitionConfig config) {

return new Predicate<Long>() {

private final int partitionID = config.getIndexingHashPartitionID();

private final int numPartitions = config.getNumPartitions();

private final EarlybirdPartitioningFunction partitioner =

new GeneralEarlybirdPartitioningFunction();

@Override

public boolean apply(Long userId) {

// See SEARCH-6675

// Right now if the partitioning logic changes in ArchivePartitioning this logic

// needs to be updated too.

return partitioner.getPartition(userId, numPartitions) == partitionID;

}

};

}

}