package com.twitter.simclusters\_v2.common.clustering

import com.twitter.simclusters\_v2.common.UserId

import com.twitter.simclusters\_v2.thriftscala.NeighborWithWeights

/\*\*

\* Select a cluster member as cluster representative.

\*/

trait ClusterRepresentativeSelectionMethod[T] {

/\*\*

\* The main external-facing method. Sub-classes should implement this method.

\*

\* @param cluster A set of NeighborWithWeights.

\* @param embeddings A map of producer ID -> embedding.

\*

\* @return UserId of the member chosen as representative.

\*/

def selectClusterRepresentative(

cluster: Set[NeighborWithWeights],

embeddings: Map[UserId, T]

): UserId

}

object ClusterRepresentativeSelectionStatistics {

// Statistics, to be imported where recorded.

val StatClusterRepresentativeSelectionTime = "cluster\_representative\_selection\_total\_time\_ms"

}