package com.twitter.simclusters\_v2.summingbird.stores

import com.twitter.frigate.common.store.strato.StratoStore

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

import com.twitter.simclusters\_v2.common.ModelVersions.\_

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

EmbeddingType,

InternalId,

LocaleEntityId,

SimClustersEmbeddingId,

SimClustersEmbedding => ThriftSimClustersEmbedding

}

import com.twitter.storehaus.ReadableStore

import com.twitter.strato.client.Client

import com.twitter.strato.thrift.ScroogeConvImplicits.\_

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

/\*\*

\* entity -> List< cluster >

\*/

object SemanticCoreEntityEmbeddingStore {

private val column =

"recommendations/simclusters\_v2/embeddings/semanticCoreEntityPerLanguageEmbeddings20M145KUpdated"

/\*\*

\* Default store, wrapped in generic data types. Use this if you know the underlying key struct.

\*/

private def getDefaultStore(

stratoClient: Client

): ReadableStore[SimClustersEmbeddingId, ThriftSimClustersEmbedding] = {

StratoStore

.withUnitView[SimClustersEmbeddingId, ThriftSimClustersEmbedding](stratoClient, column)

}

def getFavBasedLocaleEntityEmbeddingStore(

stratoClient: Client

): ReadableStore[LocaleEntityId, SimClustersEmbedding] = {

getDefaultStore(stratoClient)

.composeKeyMapping[LocaleEntityId] { entityId =>

SimClustersEmbeddingId(

EmbeddingType.FavBasedSematicCoreEntity,

ModelVersions.Model20M145KUpdated,

InternalId.LocaleEntityId(entityId)

)

}

.mapValues(SimClustersEmbedding(\_))

}

}