package com.twitter.home\_mixer.candidate\_pipeline

import com.twitter.home\_mixer.functional\_component.candidate\_source.StaleTweetsCacheCandidateSource

import com.twitter.home\_mixer.functional\_component.decorator.urt.builder.HomeFeedbackActionInfoBuilder

import com.twitter.home\_mixer.functional\_component.feature\_hydrator.NamesFeatureHydrator

import com.twitter.home\_mixer.functional\_component.query\_transformer.EditedTweetsCandidatePipelineQueryTransformer

import com.twitter.home\_mixer.service.HomeMixerAlertConfig

import com.twitter.product\_mixer.component\_library.decorator.urt.UrtItemCandidateDecorator

import com.twitter.product\_mixer.component\_library.decorator.urt.builder.contextual\_ref.ContextualTweetRefBuilder

import com.twitter.product\_mixer.component\_library.decorator.urt.builder.item.tweet.TweetCandidateUrtItemBuilder

import com.twitter.product\_mixer.component\_library.decorator.urt.builder.metadata.EmptyClientEventInfoBuilder

import com.twitter.product\_mixer.component\_library.model.candidate.TweetCandidate

import com.twitter.product\_mixer.core.functional\_component.candidate\_source.BaseCandidateSource

import com.twitter.product\_mixer.core.functional\_component.decorator.CandidateDecorator

import com.twitter.product\_mixer.core.functional\_component.feature\_hydrator.BaseCandidateFeatureHydrator

import com.twitter.product\_mixer.core.functional\_component.transformer.CandidatePipelineQueryTransformer

import com.twitter.product\_mixer.core.functional\_component.transformer.CandidatePipelineResultsTransformer

import com.twitter.product\_mixer.core.model.common.identifier.CandidatePipelineIdentifier

import com.twitter.product\_mixer.core.model.marshalling.response.rtf.safety\_level.TimelineFocalTweetSafetyLevel

import com.twitter.product\_mixer.core.model.marshalling.response.urt.contextual\_ref.TweetHydrationContext

import com.twitter.product\_mixer.core.model.marshalling.response.urt.item.tweet.TweetItem

import com.twitter.product\_mixer.core.pipeline.PipelineQuery

import com.twitter.product\_mixer.core.pipeline.candidate.DependentCandidatePipelineConfig

import javax.inject.Inject

import javax.inject.Singleton

/\*\*

\* Candidate Pipeline Config that fetches edited tweets from the Stale Tweets Cache

\*/

@Singleton

case class EditedTweetsCandidatePipelineConfig @Inject() (

staleTweetsCacheCandidateSource: StaleTweetsCacheCandidateSource,

namesFeatureHydrator: NamesFeatureHydrator,

homeFeedbackActionInfoBuilder: HomeFeedbackActionInfoBuilder)

extends DependentCandidatePipelineConfig[

PipelineQuery,

Seq[Long],

Long,

TweetCandidate

] {

override val identifier: CandidatePipelineIdentifier = CandidatePipelineIdentifier("EditedTweets")

override val candidateSource: BaseCandidateSource[Seq[Long], Long] =

staleTweetsCacheCandidateSource

override val queryTransformer: CandidatePipelineQueryTransformer[

PipelineQuery,

Seq[Long]

] = EditedTweetsCandidatePipelineQueryTransformer

override val resultTransformer: CandidatePipelineResultsTransformer[

Long,

TweetCandidate

] = { candidate => TweetCandidate(id = candidate) }

override val postFilterFeatureHydration: Seq[

BaseCandidateFeatureHydrator[PipelineQuery, TweetCandidate, \_]

] = Seq(namesFeatureHydrator)

override val decorator: Option[CandidateDecorator[PipelineQuery, TweetCandidate]] = {

val tweetItemBuilder = TweetCandidateUrtItemBuilder[PipelineQuery, TweetCandidate](

clientEventInfoBuilder = EmptyClientEventInfoBuilder,

entryIdToReplaceBuilder = Some((\_, candidate, \_) =>

Some(s"${TweetItem.TweetEntryNamespace}-${candidate.id.toString}")),

contextualTweetRefBuilder = Some(

ContextualTweetRefBuilder(

TweetHydrationContext(

// Apply safety level that includes canonical VF treatments that apply regardless of context.

safetyLevelOverride = Some(TimelineFocalTweetSafetyLevel),

outerTweetContext = None

)

)

),

feedbackActionInfoBuilder = Some(homeFeedbackActionInfoBuilder)

)

Some(UrtItemCandidateDecorator(tweetItemBuilder))

}

override val alerts = Seq(

HomeMixerAlertConfig.BusinessHours.defaultSuccessRateAlert(99.5, 50, 60, 60)

)

}