package com.twitter.product\_mixer.component\_library.decorator.urt.builder.item.unified\_trend\_event

import com.twitter.product\_mixer.component\_library.decorator.urt.builder.item.event\_summary.EventCandidateUrtItemBuilder

import com.twitter.product\_mixer.component\_library.decorator.urt.builder.item.trend.TrendCandidateUrtItemBuilder

import com.twitter.product\_mixer.component\_library.model.candidate.trends\_events.UnifiedEventCandidate

import com.twitter.product\_mixer.component\_library.model.candidate.trends\_events.UnifiedTrendCandidate

import com.twitter.product\_mixer.component\_library.model.candidate.trends\_events.UnifiedTrendEventCandidate

import com.twitter.product\_mixer.core.feature.featuremap.FeatureMap

import com.twitter.product\_mixer.core.functional\_component.decorator.urt.builder.CandidateUrtEntryBuilder

import com.twitter.product\_mixer.core.model.marshalling.response.urt.TimelineItem

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

case class UnifiedTrendEventCandidateUrtItemBuilder[Query <: PipelineQuery](

eventCandidateUrtItemBuilder: EventCandidateUrtItemBuilder[Query],

trendCandidateUrtItemBuilder: TrendCandidateUrtItemBuilder[Query])

extends CandidateUrtEntryBuilder[Query, UnifiedTrendEventCandidate[Any], TimelineItem] {

override def apply(

query: Query,

candidate: UnifiedTrendEventCandidate[Any],

candidateFeatures: FeatureMap

): TimelineItem = {

candidate match {

case event: UnifiedEventCandidate =>

eventCandidateUrtItemBuilder(

query = query,

candidate = event,

candidateFeatures = candidateFeatures)

case trend: UnifiedTrendCandidate =>

trendCandidateUrtItemBuilder(

query = query,

candidate = trend,

candidateFeatures = candidateFeatures)

}

}

}