package com.twitter.product\_mixer.component\_library.decorator.urt.builder.richtext

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

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

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

import com.twitter.product\_mixer.core.model.common.UniversalNoun

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

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

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

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

case class RichTextBuilder[-Query <: PipelineQuery, -Candidate <: UniversalNoun[Any]](

textBuilder: BaseStr[Query, Candidate],

linkMap: Map[String, String],

rtl: Option[Boolean],

alignment: Option[RichTextAlignment],

linkTypeMap: Map[String, UrlType] = Map.empty)

extends BaseRichTextBuilder[Query, Candidate] {

def apply(query: Query, candidate: Candidate, candidateFeatures: FeatureMap): RichText = {

RichTextMarkupUtil.richTextFromMarkup(

text = textBuilder(query, candidate, candidateFeatures),

linkMap = linkMap,

rtl = rtl,

alignment = alignment,

linkTypeMap = linkTypeMap)

}

}