package com.twitter.product\_mixer.component\_library.premarshaller.urt.builder

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

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

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

import com.twitter.stringcenter.client.StringCenter

import com.twitter.stringcenter.client.core.ExternalString

trait BaseUrtMetadataBuilder[-Query <: PipelineQuery] {

def build(

query: Query,

entries: Seq[TimelineEntry]

): TimelineMetadata

}

case class UrtMetadataBuilder(

title: Option[String] = None,

scribeConfigBuilder: Option[TimelineScribeConfigBuilder[PipelineQuery]])

extends BaseUrtMetadataBuilder[PipelineQuery] {

override def build(

query: PipelineQuery,

entries: Seq[TimelineEntry]

): TimelineMetadata = TimelineMetadata(

title = title,

scribeConfig = scribeConfigBuilder.flatMap(\_.build(query, entries))

)

}

case class UrtMetadataStringCenterBuilder(

titleKey: ExternalString,

scribeConfigBuilder: Option[TimelineScribeConfigBuilder[PipelineQuery]],

stringCenter: StringCenter)

extends BaseUrtMetadataBuilder[PipelineQuery] {

override def build(

query: PipelineQuery,

entries: Seq[TimelineEntry]

): TimelineMetadata = TimelineMetadata(

title = Some(stringCenter.prepare(titleKey)),

scribeConfig = scribeConfigBuilder.flatMap(\_.build(query, entries))

)

}