package com.twitter.product\_mixer.core.functional\_component.marshaller.response.urt.media

import com.twitter.product\_mixer.core.functional\_component.marshaller.response.urt.metadata.ImageVariantMarshaller

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

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

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

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

import com.twitter.timelines.render.{thriftscala => urt}

import javax.inject.Inject

import javax.inject.Singleton

@Singleton

class MediaEntityMarshaller @Inject() (

tweetMediaMarshaller: TweetMediaMarshaller,

broadcastIdMarshaller: BroadcastIdMarshaller,

imageVariantMarshaller: ImageVariantMarshaller) {

def apply(mediaEntity: MediaEntity): urt.MediaEntity = mediaEntity match {

case tweetMedia: TweetMedia => urt.MediaEntity.TweetMedia(tweetMediaMarshaller(tweetMedia))

case broadcastId: BroadcastId => urt.MediaEntity.BroadcastId(broadcastIdMarshaller(broadcastId))

case image: Image => urt.MediaEntity.Image(imageVariantMarshaller(image.image))

}

}