package com.twitter.product\_mixer.core.functional\_component.marshaller.response.urt.item.generic\_summary\_item

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

import com.twitter.product\_mixer.core.functional\_component.marshaller.response.urt.promoted.PromotedMetadataMarshaller

import com.twitter.product\_mixer.core.functional\_component.marshaller.response.urt.richtext.RichTextMarshaller

import com.twitter.product\_mixer.core.model.marshalling.response.urt.item.generic\_summary.GenericSummaryItem

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

import javax.inject.Inject

import javax.inject.Singleton

@Singleton

class GenericSummaryItemMarshaller @Inject() (

genericSummaryDisplayTypeMarshaller: GenericSummaryDisplayTypeMarshaller,

genericSummaryContextMarshaller: GenericSummaryContextMarshaller,

genericSummaryActionMarshaller: GenericSummaryActionMarshaller,

mediaMarshaller: MediaMarshaller,

promotedMetadataMarshaller: PromotedMetadataMarshaller,

richTextMarshaller: RichTextMarshaller) {

def apply(genericSummaryItem: GenericSummaryItem): urt.TimelineItemContent =

urt.TimelineItemContent.GenericSummary(

urt.GenericSummary(

headline = richTextMarshaller(genericSummaryItem.headline),

displayType = genericSummaryDisplayTypeMarshaller(genericSummaryItem.displayType),

userAttributionIds = genericSummaryItem.userAttributionIds,

media = genericSummaryItem.media.map(mediaMarshaller(\_)),

context = genericSummaryItem.context.map(genericSummaryContextMarshaller(\_)),

timestamp = genericSummaryItem.timestamp.map(\_.inMilliseconds),

onClickAction = genericSummaryItem.onClickAction.map(genericSummaryActionMarshaller(\_)),

promotedMetadata = genericSummaryItem.promotedMetadata.map(promotedMetadataMarshaller(\_))

)

)

}