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

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

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

import com.twitter.product\_mixer.core.model.marshalling.response.urt.item.trend.TrendItem

import com.twitter.timelines.render.thriftscala.GroupedTrend

import com.twitter.timelines.render.thriftscala.TrendMetadata

import javax.inject.Inject

import javax.inject.Singleton

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

@Singleton

class TrendItemMarshaller @Inject() (

promotedMetadataMarshaller: PromotedMetadataMarshaller,

urlMarshaller: UrlMarshaller) {

def apply(trendItem: TrendItem): urt.TimelineItemContent =

urt.TimelineItemContent.Trend(

urt.Trend(

name = trendItem.trendName,

url = urlMarshaller(trendItem.url),

promotedMetadata = trendItem.promotedMetadata.map(promotedMetadataMarshaller(\_)),

description = trendItem.description,

trendMetadata = Some(

TrendMetadata(

metaDescription = trendItem.metaDescription,

url = Some(urlMarshaller(trendItem.url)),

domainContext = trendItem.domainContext

)),

groupedTrends = trendItem.groupedTrends.map { trends =>

trends.map { trend =>

GroupedTrend(name = trend.trendName, url = urlMarshaller(trend.url))

}

}

)

)

}