package com.twitter.unified\_user\_actions.adapter.ads\_callback\_engagements

import com.twitter.finagle.stats.NullStatsReceiver

import com.twitter.finagle.stats.StatsReceiver

import com.twitter.finatra.kafka.serde.UnKeyed

import com.twitter.unified\_user\_actions.adapter.AbstractAdapter

import com.twitter.ads.spendserver.thriftscala.SpendServerEvent

import com.twitter.unified\_user\_actions.thriftscala.UnifiedUserAction

class AdsCallbackEngagementsAdapter

extends AbstractAdapter[SpendServerEvent, UnKeyed, UnifiedUserAction] {

import AdsCallbackEngagementsAdapter.\_

override def adaptOneToKeyedMany(

input: SpendServerEvent,

statsReceiver: StatsReceiver = NullStatsReceiver

): Seq[(UnKeyed, UnifiedUserAction)] =

adaptEvent(input).map { e => (UnKeyed, e) }

}

object AdsCallbackEngagementsAdapter {

def adaptEvent(input: SpendServerEvent): Seq[UnifiedUserAction] = {

val baseEngagements: Seq[BaseAdsCallbackEngagement] =

EngagementTypeMappings.getEngagementMappings(Option(input).flatMap(\_.engagementEvent))

baseEngagements.flatMap(\_.getUUA(input))

}

}