package com.twitter.unified\_user\_actions.adapter.uua\_aggregates

import com.twitter.unified\_user\_actions.adapter.common.AdapterUtils

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

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

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

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

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

abstract class BaseUuaAction(actionType: ActionType) {

def getRekeyedUUA(input: UnifiedUserAction): Option[KeyedUuaTweet] =

getTweetIdFromItem(input.item).map { tweetId =>

KeyedUuaTweet(

tweetId = tweetId,

actionType = input.actionType,

userIdentifier = input.userIdentifier,

eventMetadata = EventMetadata(

sourceTimestampMs = input.eventMetadata.sourceTimestampMs,

receivedTimestampMs = AdapterUtils.currentTimestampMs,

sourceLineage = input.eventMetadata.sourceLineage

)

)

}

protected def getTweetIdFromItem(item: Item): Option[Long] = {

item match {

case Item.TweetInfo(tweetInfo) => Some(tweetInfo.actionTweetId)

case \_ => None

}

}

}

/\*\*

\* When there is a new user creation event in Gizmoduck

\*/

object ClientTweetRenderImpressionUua extends BaseUuaAction(ActionType.ClientTweetRenderImpression)