package com.twitter.unified\_user\_actions.adapter.client\_event

import com.twitter.clientapp.thriftscala.AmplifyDetails

import com.twitter.clientapp.thriftscala.MediaDetails

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

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

import com.twitter.video.analytics.thriftscala.MediaIdentifier

object VideoClientEventUtils {

/\*\*

\* For Tweets with multiple videos, find the id of the video that generated the client-event

\*/

def videoIdFromMediaIdentifier(mediaIdentifier: MediaIdentifier): Option[String] =

mediaIdentifier match {

case MediaIdentifier.MediaPlatformIdentifier(mediaPlatformIdentifier) =>

mediaPlatformIdentifier.mediaId.map(\_.toString)

case \_ => None

}

/\*\*

\* Given:

\* 1. the id of the video (`mediaId`)

\* 2. details about all the media items in the Tweet (`mediaItems`),

\* iterate over the `mediaItems` to lookup the metadata about the video with id `mediaId`.

\*/

def getVideoMetadata(

mediaId: String,

mediaItems: Seq[MediaDetails],

amplifyDetails: Option[AmplifyDetails]

): Option[TweetActionInfo] = {

mediaItems.collectFirst {

case media if media.contentId.contains(mediaId) =>

TweetActionInfo.TweetVideoWatch(

TweetVideoWatch(

mediaType = media.mediaType,

isMonetizable = media.dynamicAds,

videoType = amplifyDetails.flatMap(\_.videoType)

))

}

}

}