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

import com.twitter.tweetypie.thriftscala.TweetEventFlags

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.UnifiedUserAction

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

/\*\*

\* Base class for Tweetypie Tweet Event.

\* Extends this class if you need to implement the parser for a new Tweetypie Tweet Event Type.

\* @see https://sourcegraph.twitter.biz/git.twitter.biz/source/-/blob/src/thrift/com/twitter/tweetypie/tweet\_events.thrift?L225

\*/

trait BaseTweetypieTweetEvent[T] {

/\*\*

\* Returns an Optional UnifiedUserAction from the event.

\*/

def getUnifiedUserAction(event: T, flags: TweetEventFlags): Option[UnifiedUserAction]

/\*\*

\* Returns UnifiedUserAction.ActionType for each type of event.

\*/

protected def actionType: ActionType

/\*\*

\* Output type of the predicate. Could be an input of getItem.

\*/

type ExtractedEvent

/\*\*

\* Returns Some(ExtractedEvent) if the event is valid and None otherwise.

\*/

protected def extract(event: T): Option[ExtractedEvent]

/\*\*

\* Get the UnifiedUserAction.Item from the event.

\*/

protected def getItem(extractedEvent: ExtractedEvent, event: T): Item

/\*\*

\* Get the UnifiedUserAction.UserIdentifier from the event.

\*/

protected def getUserIdentifier(event: T): UserIdentifier

/\*\*

\* Get UnifiedUserAction.EventMetadata from the event.

\*/

protected def getEventMetadata(event: T, flags: TweetEventFlags): EventMetadata

}