package com.twitter.tweetypie

package hydrator

import com.twitter.tweetypie.core.ValueState

import com.twitter.tweetypie.thriftscala.MentionEntity

import com.twitter.tweetypie.unmentions.thriftscala.UnmentionData

object UnmentionDataHydrator {

type Type = ValueHydrator[Option[UnmentionData], Ctx]

case class Ctx(

conversationId: Option[TweetId],

mentions: Seq[MentionEntity],

underlyingTweetCtx: TweetCtx)

extends TweetCtx.Proxy

def apply(): Type = {

ValueHydrator.map[Option[UnmentionData], Ctx] { (\_, ctx) =>

val mentionedUserIds: Seq[UserId] = ctx.mentions.flatMap(\_.userId)

ValueState.modified(

Some(UnmentionData(ctx.conversationId, Option(mentionedUserIds).filter(\_.nonEmpty)))

)

}

}.onlyIf { (\_, ctx) =>

ctx.tweetFieldRequested(Tweet.UnmentionDataField)

}

}