package com.twitter.tweetypie

package hydrator

import com.twitter.tweetypie.core.\_

import com.twitter.tweetypie.media.Media

import com.twitter.tweetypie.media.MediaUrl

import com.twitter.tweetypie.thriftscala.\_

object MediaUrlFieldsHydrator {

type Ctx = MediaEntityHydrator.Cacheable.Ctx

type Type = MediaEntityHydrator.Cacheable.Type

def mediaPermalink(ctx: Ctx): Option[UrlEntity] =

ctx.urlEntities.view.reverse.find(MediaUrl.Permalink.hasTweetId(\_, ctx.tweetId))

def apply(): Type =

ValueHydrator

.map[MediaEntity, Ctx] { (curr, ctx) =>

mediaPermalink(ctx) match {

case None => ValueState.unmodified(curr)

case Some(urlEntity) => ValueState.modified(Media.copyFromUrlEntity(curr, urlEntity))

}

}

.onlyIf((curr, ctx) => curr.url == null && Media.isOwnMedia(ctx.tweetId, curr))

}