package com.twitter.timelineranker.observe

import com.twitter.timelines.authorization.ReadRequest

import com.twitter.timelines.model.UserId

import com.twitter.timelines.observe.ObservedAndValidatedRequests

import com.twitter.timelines.observe.ServiceObserver

import com.twitter.timelines.observe.ServiceTracer

import com.twitter.util.Future

trait ObservedRequests extends ObservedAndValidatedRequests {

def observeAndValidate[R, Q](

request: Q,

viewerIds: Seq[UserId],

stats: ServiceObserver.Stats[Q],

exceptionHandler: PartialFunction[Throwable, Future[R]]

)(

f: Q => Future[R]

): Future[R] = {

super.observeAndValidate[Q, R](

request,

viewerIds,

ReadRequest,

validateRequest,

exceptionHandler,

stats,

ServiceTracer.identity[Q]

)(f)

}

def validateRequest[Q](request: Q): Unit = {

// TimelineQuery and its derived classes do not permit invalid instances to be constructed.

// Therefore no additional validation is required.

}

}