package com.twitter.timelineranker.model

import com.twitter.timelineranker.{thriftscala => thrift}

import com.twitter.util.Time

object TimeRange {

val default: TimeRange = TimeRange(None, None)

def fromThrift(range: thrift.TimeRange): TimeRange = {

TimeRange(

from = range.fromMs.map(Time.fromMilliseconds),

to = range.toMs.map(Time.fromMilliseconds)

)

}

}

case class TimeRange(from: Option[Time], to: Option[Time]) extends TimelineRange {

throwIfInvalid()

def throwIfInvalid(): Unit = {

(from, to) match {

case (Some(fromTime), Some(toTime)) =>

require(fromTime <= toTime, "from-time must be less than or equal to-time.")

case \_ => // valid, do nothing.

}

}

def toThrift: thrift.TimeRange = {

thrift.TimeRange(

fromMs = from.map(\_.inMilliseconds),

toMs = to.map(\_.inMilliseconds)

)

}

def toTimelineRangeThrift: thrift.TimelineRange = {

thrift.TimelineRange.TimeRange(toThrift)

}

}