package com.twitter.product\_mixer.component\_library.model.cursor

import com.twitter.product\_mixer.core.pipeline.PipelineCursor

import com.twitter.product\_mixer.core.pipeline.UrtPipelineCursor

import com.twitter.product\_mixer.core.model.marshalling.response.slice.CursorType

import com.twitter.product\_mixer.core.model.marshalling.response.urt.operation.{

CursorType => UrtCursorType

}

/\*\*

\* Cursor model that may be used when cursoring over an ordered candidate source.

\*

\* @param initialSortIndex See [[UrtPipelineCursor]]

\* @param id represents the ID of the element, typically the top element for a top cursor or the

\* bottom element for a bottom cursor, in an ordered candidate list

\* @param gapBoundaryId represents the ID of the gap boundary element, which in gap cursors is the

\* opposite bound of the gap to be filled with the cursor

\*/

case class UrtOrderedCursor(

override val initialSortIndex: Long,

id: Option[Long],

cursorType: Option[UrtCursorType],

gapBoundaryId: Option[Long] = None)

extends UrtPipelineCursor

case class OrderedCursor(

id: Option[Long],

cursorType: Option[CursorType],

gapBoundaryId: Option[Long] = None)

extends PipelineCursor