package com.twitter.product\_mixer.core.pipeline

import com.twitter.product\_mixer.core.pipeline.pipeline\_failure.MalformedCursor

import com.twitter.product\_mixer.core.pipeline.pipeline\_failure.PipelineFailure

import com.twitter.scrooge.BinaryThriftStructSerializer

import com.twitter.scrooge.ThriftStruct

import com.twitter.util.Return

import com.twitter.util.Throw

import com.twitter.util.Try

/\*\*

\* Serializes a [[PipelineCursor]] into thrift and then into a base64 encoded string

\*/

trait PipelineCursorSerializer[-Cursor <: PipelineCursor] {

def serializeCursor(cursor: Cursor): String

}

object PipelineCursorSerializer {

/\*\*

\* Deserializes a cursor string into thrift and then into a [[PipelineCursor]]

\*

\* @param cursorString to deserialize, which is base64 encoded thrift

\* @param cursorThriftSerializer to deserialize the cursor string into thrift

\* @param deserializePf specifies how to transform the serialized thrift into a [[PipelineCursor]]

\* @return optional [[PipelineCursor]]. `None` may or may not be a failure depending on the

\* implementation of deserializePf.

\*

\* @note The "A" type of deserializePf cannot be inferred due to the thrift type not being present

\* on the PipelineCursorSerializer trait. Therefore invokers must often add an explicit type

\* on the deserializeCursor call to help out the compiler when passing deserializePf inline.

\* Alternatively, deserializePf can be declared as a val with a type annotation before it is

\* passed into this method.

\*/

def deserializeCursor[Thrift <: ThriftStruct, Cursor <: PipelineCursor](

cursorString: String,

cursorThriftSerializer: BinaryThriftStructSerializer[Thrift],

deserializePf: PartialFunction[Option[Thrift], Option[Cursor]]

): Option[Cursor] = {

val thriftCursor: Option[Thrift] =

Try {

cursorThriftSerializer.fromString(cursorString)

} match {

case Return(thriftCursor) => Some(thriftCursor)

case Throw(\_) => None

}

// Add type annotation to help out the compiler since the type is lost due to the \_ match

val defaultDeserializePf: PartialFunction[Option[Thrift], Option[Cursor]] = {

case \_ =>

// This case is the result of the client submitting a cursor we do not expect

throw PipelineFailure(MalformedCursor, s"Unknown request cursor: $cursorString")

}

(deserializePf orElse defaultDeserializePf)(thriftCursor)

}

}