package com.twitter.product\_mixer.component\_library.premarshaller.urt.builder

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

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

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

import com.twitter.product\_mixer.core.model.marshalling.response.urt.metadata.MarkUnreadableEntry

/\*\*

\* Build a MarkUnreadEntries instruction

\*

\* Note that this implementation currently supports top-level entries, but not module item entries.

\*/

case class MarkUnreadInstructionBuilder[Query <: PipelineQuery](

override val includeInstruction: IncludeInstruction[Query] = AlwaysInclude)

extends UrtInstructionBuilder[Query, MarkEntriesUnreadInstruction] {

override def build(

query: Query,

entries: Seq[TimelineEntry]

): Seq[MarkEntriesUnreadInstruction] = {

if (includeInstruction(query, entries)) {

val filteredEntries = entries.collect {

case entry: MarkUnreadableEntry if entry.isMarkUnread.contains(true) =>

entry.entryIdentifier

}

if (filteredEntries.nonEmpty) Seq(MarkEntriesUnreadInstruction(filteredEntries))

else Seq.empty

} else {

Seq.empty

}

}

}