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

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

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

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

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

case class AddToModuleInstructionBuilder[Query <: PipelineQuery](

override val includeInstruction: IncludeInstruction[Query] = AlwaysInclude)

extends UrtInstructionBuilder[Query, AddToModuleTimelineInstruction] {

override def build(

query: Query,

entries: Seq[TimelineEntry]

): Seq[AddToModuleTimelineInstruction] = {

if (includeInstruction(query, entries)) {

val moduleEntries = entries.collect {

case module: TimelineModule => module

}

if (moduleEntries.nonEmpty) {

assert(moduleEntries.size == 1, "Currently we only support appending to one module")

moduleEntries.headOption.map { moduleEntry =>

AddToModuleTimelineInstruction(

moduleItems = moduleEntry.items,

moduleEntryId = moduleEntry.entryIdentifier,

// Currently configuring moduleItemEntryId and prepend fields are not supported.

moduleItemEntryId = None,

prepend = None

)

}

}.toSeq

else Seq.empty

} else {

Seq.empty

}

}

}