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

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

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

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

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

case class ShowCoverInstructionBuilder[Query <: PipelineQuery](

override val includeInstruction: IncludeInstruction[Query] = AlwaysInclude)

extends UrtInstructionBuilder[Query, ShowCoverInstruction] {

override def build(

query: Query,

entries: Seq[TimelineEntry]

): Seq[ShowCoverInstruction] = {

if (includeInstruction(query, entries)) {

// Currently only one cover is supported per response

entries.collectFirst {

case coverEntry: Cover => ShowCoverInstruction(coverEntry)

}.toSeq

} else {

Seq.empty

}

}

}