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

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

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

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

/\*\*

\* Build the AddEntries instruction with special handling for replaceable entries.

\*

\* Entries (though almost always a single entry) with a non-empty entryIdToReplace field should be

\* collected and transformed into ReplaceEntry instructions. These should be filtered out of the

\* AddEntries instruction. We avoid doing this as part of the regular AddEntriesInstructionBuilder

\* because replacement is rare and detecting replaceable entries takes linear time.

\*/

case class AddEntriesWithReplaceInstructionBuilder[Query <: PipelineQuery](

override val includeInstruction: IncludeInstruction[Query] = AlwaysInclude)

extends UrtInstructionBuilder[Query, AddEntriesTimelineInstruction] {

override def build(

query: Query,

entries: Seq[TimelineEntry]

): Seq[AddEntriesTimelineInstruction] = {

if (includeInstruction(query, entries)) {

val entriesToAdd = entries.filter(\_.entryIdToReplace.isEmpty)

if (entriesToAdd.nonEmpty) Seq(AddEntriesTimelineInstruction(entriesToAdd))

else Seq.empty

} else {

Seq.empty

}

}

}