package com.twitter.product\_mixer.core.functional\_component.candidate\_source.strato

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

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

import com.twitter.stitch.Stitch

import com.twitter.strato.response.Err

/\*\*

\* Categorize Strato's Err messages to our PipelineFailures.

\*

\* This should be used by all strato-based candidate source, and we can

\* add more cases here as they're useful.

\*/

object StratoErrCategorizer {

val CategorizeStratoException: PartialFunction[Throwable, Stitch[Nothing]] = {

case err @ Err(Err.Authorization, reason, context) =>

Stitch.exception(

PipelineFailure(Unauthorized, s"$reason [${context.toString}]", underlying = Some(err))

)

}

}