package com.twitter.product\_mixer.component\_library.gate

import com.twitter.product\_mixer.core.functional\_component.gate.Gate

import com.twitter.product\_mixer.core.model.common.identifier.ComponentIdentifier

import com.twitter.product\_mixer.core.model.common.identifier.GateIdentifier

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

import com.twitter.product\_mixer.core.quality\_factor.HasQualityFactorStatus

import com.twitter.stitch.Stitch

/\*\*

\* A Gate that only continues if the quality factor value of the pipeline is above the given

\* threshold. This is useful for disabling an expensive function when the pipeline is under pressure

\* (quality factor is low).

\*/

case class QualityFactorGate(pipelineIdentifier: ComponentIdentifier, threshold: Double)

extends Gate[PipelineQuery with HasQualityFactorStatus] {

override val identifier: GateIdentifier = GateIdentifier(

s"${pipelineIdentifier.name}QualityFactor")

override def shouldContinue(

query: PipelineQuery with HasQualityFactorStatus

): Stitch[Boolean] =

Stitch.value(query.getQualityFactorCurrentValue(pipelineIdentifier) >= threshold)

}