package com.twitter.product\_mixer.core.pipeline

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

import com.twitter.product\_mixer.core.pipeline.state.HasExecutorResults

import com.twitter.product\_mixer.core.pipeline.state.HasResult

/\*\*

\* A pipeline builder that is responsible for taking a PipelineConfig and creating a final pipeline

\* from it. It provides an [[NewPipelineArrowBuilder]] for composing the pipeline's underlying arrow

\* from [[Step]]s.

\*

\* @tparam Config The Pipeline Config

\* @tparam PipelineArrowResult The expected final result

\* @tparam PipelineArrowState State object for maintaining state across the pipeline.

\* @tparam OutputPipeline The final pipeline

\*/

trait NewPipelineBuilder[

Config <: PipelineConfig,

PipelineArrowResult,

PipelineArrowState <: HasExecutorResults[PipelineArrowState] with HasResult[PipelineArrowResult],

OutputPipeline <: Pipeline[\_, \_]] {

type ArrowResult = PipelineArrowResult

type ArrowState = PipelineArrowState

def build(

parentComponentIdentifierStack: ComponentIdentifierStack,

arrowBuilder: NewPipelineArrowBuilder[ArrowResult, ArrowState],

config: Config

): OutputPipeline

}