package com.twitter.unified\_user\_actions.enricher.hydrator

import com.twitter.unified\_user\_actions.enricher.ImplementationException

import com.twitter.unified\_user\_actions.enricher.internal.thriftscala.EnrichmentEnvelop

import com.twitter.unified\_user\_actions.enricher.internal.thriftscala.EnrichmentInstruction

import com.twitter.unified\_user\_actions.enricher.internal.thriftscala.EnrichmentKey

import com.twitter.util.Future

/\*\*

\* This hydrator does nothing. If it's used by mistake for any reason, an exception will be thrown.

\* Use this when you expect to have no hydration (for example, the planner shouldn't hydrate anything

\* and only would perform the partitioning function).

\*/

object NoopHydrator {

val OutputTopic: Option[String] = None

}

class NoopHydrator extends Hydrator {

override def hydrate(

instruction: EnrichmentInstruction,

key: Option[EnrichmentKey],

envelop: EnrichmentEnvelop

): Future[EnrichmentEnvelop] = {

throw new ImplementationException(

"NoopHydrator shouldn't be invoked when configure. Check your " +

"enrichment plan.")

}

}