package com.twitter.frigate.pushservice.model.ibis

import com.twitter.frigate.pushservice.model.TopicProofTweetPushCandidate

import com.twitter.frigate.pushservice.exception.UttEntityNotFoundException

import com.twitter.util.Future

trait TopicProofTweetIbis2Hydrator extends TweetCandidateIbis2Hydrator {

self: TopicProofTweetPushCandidate =>

private lazy val implicitTopicTweetModelValues: Map[String, String] = {

val uttEntity = localizedUttEntity.getOrElse(

throw new UttEntityNotFoundException(

s"${getClass.getSimpleName} UttEntity missing for $tweetId"))

Map(

"topic\_name" -> uttEntity.localizedNameForDisplay,

"topic\_id" -> uttEntity.entityId.toString

)

}

override lazy val modelName: String = pushCopy.ibisPushModelName

override lazy val tweetModelValues: Future[Map[String, String]] =

for {

superModelValues <- super.tweetModelValues

tweetInlineModelValues <- tweetInlineActionModelValue

} yield {

superModelValues ++

tweetInlineModelValues ++

implicitTopicTweetModelValues

}

}