package com.twitter.cr\_mixer.param

import com.twitter.timelines.configapi.BaseConfig

import com.twitter.timelines.configapi.BaseConfigBuilder

import com.twitter.timelines.configapi.FSName

import com.twitter.timelines.configapi.FSParam

import com.twitter.timelines.configapi.FeatureSwitchOverrideUtil

import com.twitter.timelines.configapi.Param

/\*\*

\* ConsumersBasedUserTweetGraph Params, there are multiple ways (e.g. FRS, RealGraphOon) to generate consumersSeedSet for ConsumersBasedUserTweetGraph

\* for now we allow flexibility in tuning UTG params for different consumersSeedSet generation algo by giving the param name {consumerSeedSetAlgo}{ParamName}

\*/

object ConsumersBasedUserTweetGraphParams {

object EnableSourceParam

extends FSParam[Boolean](

name = "consumers\_based\_user\_tweet\_graph\_enable\_source",

default = false

)

val AllParams: Seq[Param[\_] with FSName] = Seq(

EnableSourceParam,

)

lazy val config: BaseConfig = {

val intOverrides = FeatureSwitchOverrideUtil.getBoundedIntFSOverrides()

val doubleOverrides =

FeatureSwitchOverrideUtil.getBoundedDoubleFSOverrides()

val booleanOverrides = FeatureSwitchOverrideUtil.getBooleanFSOverrides(

EnableSourceParam

)

BaseConfigBuilder()

.set(intOverrides: \_\*)

.set(booleanOverrides: \_\*)

.set(doubleOverrides: \_\*)

.build()

}

}