package com.twitter.cr\_mixer.param

import com.twitter.conversions.DurationOps.\_

import com.twitter.finagle.stats.NullStatsReceiver

import com.twitter.logging.Logger

import com.twitter.simclusters\_v2.common.ModelVersions

import com.twitter.timelines.configapi.BaseConfig

import com.twitter.timelines.configapi.BaseConfigBuilder

import com.twitter.timelines.configapi.DurationConversion

import com.twitter.timelines.configapi.FSBoundedParam

import com.twitter.timelines.configapi.FSEnumParam

import com.twitter.timelines.configapi.FSName

import com.twitter.timelines.configapi.FeatureSwitchOverrideUtil

import com.twitter.timelines.configapi.HasDurationConversion

import com.twitter.timelines.configapi.Param

import com.twitter.util.Duration

/\*\*

\* Instantiate Params that do not relate to a specific product.

\* The params in this file correspond to config repo file

\* [[https://sourcegraph.twitter.biz/config-git.twitter.biz/config/-/blob/features/cr-mixer/main/twistly\_core.yml]]

\*/

object GlobalParams {

object MaxCandidatesPerRequestParam

extends FSBoundedParam[Int](

name = "twistly\_core\_max\_candidates\_per\_request",

default = 100,

min = 0,

max = 9000

)

object ModelVersionParam

extends FSEnumParam[ModelVersions.Enum.type](

name = "twistly\_core\_simclusters\_model\_version\_id",

default = ModelVersions.Enum.Model20M145K2020,

enum = ModelVersions.Enum

)

object UnifiedMaxSourceKeyNum

extends FSBoundedParam[Int](

name = "twistly\_core\_unified\_max\_sourcekey\_num",

default = 15,

min = 0,

max = 100

)

object MaxCandidateNumPerSourceKeyParam

extends FSBoundedParam[Int](

name = "twistly\_core\_candidate\_per\_sourcekey\_max\_num",

default = 200,

min = 0,

max = 1000

)

// 1 hours to 30 days

object MaxTweetAgeHoursParam

extends FSBoundedParam[Duration](

name = "twistly\_core\_max\_tweet\_age\_hours",

default = 720.hours,

min = 1.hours,

max = 720.hours

)

with HasDurationConversion {

override val durationConversion: DurationConversion = DurationConversion.FromHours

}

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

MaxCandidatesPerRequestParam,

UnifiedMaxSourceKeyNum,

MaxCandidateNumPerSourceKeyParam,

ModelVersionParam,

MaxTweetAgeHoursParam

)

lazy val config: BaseConfig = {

val booleanOverrides = FeatureSwitchOverrideUtil.getBooleanFSOverrides()

val intOverrides = FeatureSwitchOverrideUtil.getBoundedIntFSOverrides(

MaxCandidatesPerRequestParam,

UnifiedMaxSourceKeyNum,

MaxCandidateNumPerSourceKeyParam

)

val enumOverrides = FeatureSwitchOverrideUtil.getEnumFSOverrides(

NullStatsReceiver,

Logger(getClass),

ModelVersionParam

)

val boundedDurationFSOverrides =

FeatureSwitchOverrideUtil.getBoundedDurationFSOverrides(MaxTweetAgeHoursParam)

val seqOverrides = FeatureSwitchOverrideUtil.getLongSeqFSOverrides()

BaseConfigBuilder()

.set(booleanOverrides: \_\*)

.set(intOverrides: \_\*)

.set(boundedDurationFSOverrides: \_\*)

.set(enumOverrides: \_\*)

.set(seqOverrides: \_\*)

.build()

}

}