package com.twitter.timelineranker.parameters.in\_network\_tweets

import com.twitter.timelineranker.parameters.recap.RecapQueryContext

import com.twitter.timelines.configapi.decider.\_

import com.twitter.timelines.configapi.FSBoundedParam

import com.twitter.timelines.configapi.FSParam

import com.twitter.timelines.configapi.Param

object InNetworkTweetParams {

import RecapQueryContext.\_

/\*\*

\* Controls limit on the number of followed users fetched from SGS.

\*

\* The specific default value below is for blender-timelines parity.

\*/

object MaxFollowedUsersParam

extends FSBoundedParam[Int](

name = "recycled\_max\_followed\_users",

default = MaxFollowedUsers.default,

min = MaxFollowedUsers.bounds.minInclusive,

max = MaxFollowedUsers.bounds.maxInclusive

)

/\*\*

\* Controls limit on the number of hits for Earlybird.

\*

\*/

object RelevanceOptionsMaxHitsToProcessParam

extends FSBoundedParam[Int](

name = "recycled\_relevance\_options\_max\_hits\_to\_process",

default = 500,

min = 100,

max = 20000

)

/\*\*

\* Fallback value for maximum number of search results, if not specified by query.maxCount

\*/

object DefaultMaxTweetCount extends Param(200)

/\*\*

\* We multiply maxCount (caller supplied value) by this multiplier and fetch those many

\* candidates from search so that we are left with sufficient number of candidates after

\* hydration and filtering.

\*/

object MaxCountMultiplierParam

extends Param(MaxCountMultiplier.default)

with DeciderValueConverter[Double] {

override def convert: IntConverter[Double] =

OutputBoundIntConverter[Double](divideDeciderBy100 \_, MaxCountMultiplier.bounds)

}

/\*\*

\* Enable [[SearchQueryBuilder.createExcludedSourceTweetIdsQuery]]

\*/

object EnableExcludeSourceTweetIdsQueryParam

extends FSParam[Boolean](

name = "recycled\_exclude\_source\_tweet\_ids\_query\_enable",

default = false

)

object EnableEarlybirdReturnAllResultsParam

extends FSParam[Boolean](

name = "recycled\_enable\_earlybird\_return\_all\_results",

default = true

)

/\*\*

\* FS-controlled param to enable anti-dilution transform for DDG-16198

\*/

object RecycledMaxFollowedUsersEnableAntiDilutionParam

extends FSParam[Boolean](

name = "recycled\_max\_followed\_users\_enable\_anti\_dilution",

default = false

)

/\*\*

\* Enables semantic core, penguin, and tweetypie content features in recycled source.

\*/

object EnableContentFeaturesHydrationParam extends Param(default = true)

/\*\*

\* additionally enables tokens when hydrating content features.

\*/

object EnableTokensInContentFeaturesHydrationParam

extends FSParam(

name = "recycled\_enable\_tokens\_in\_content\_features\_hydration",

default = false

)

/\*\*

\* additionally enables tweet text when hydrating content features.

\* This only works if EnableContentFeaturesHydrationParam is set to true

\*/

object EnableTweetTextInContentFeaturesHydrationParam

extends FSParam(

name = "recycled\_enable\_tweet\_text\_in\_content\_features\_hydration",

default = false

)

/\*\*

\* Enables hydrating root tweet of in-network replies and extended replies

\*/

object EnableReplyRootTweetHydrationParam

extends FSParam(

name = "recycled\_enable\_reply\_root\_tweet\_hydration",

default = true

)

/\*\*

\* additionally enables conversationControl when hydrating content features.

\* This only works if EnableContentFeaturesHydrationParam is set to true

\*/

object EnableConversationControlInContentFeaturesHydrationParam

extends FSParam(

name = "conversation\_control\_in\_content\_features\_hydration\_recycled\_enable",

default = false

)

object EnableTweetMediaHydrationParam

extends FSParam(

name = "tweet\_media\_hydration\_recycled\_enable",

default = false

)

object EnableEarlybirdRealtimeCgMigrationParam

extends FSParam(

name = "recycled\_enable\_earlybird\_realtime\_cg\_migration",

default = false

)

}