package com.twitter.frigate.pushservice.util

import com.twitter.frigate.thriftscala.CommonRecommendationType

import com.twitter.frigate.thriftscala.CommonRecommendationType.\_

object IbisScribeTargets {

val User2 = "magic\_rec\_user\_2"

val User4 = "magic\_rec\_user\_4"

val Tweet2 = "magic\_rec\_tweet\_2"

val Tweet4 = "magic\_rec\_tweet\_4"

val Tweet5 = "magic\_rec\_tweet\_5"

val Tweet9 = "magic\_rec\_tweet\_9"

val Tweet10 = "magic\_rec\_tweet\_10"

val Tweet11 = "magic\_rec\_tweet\_11"

val Tweet12 = "magic\_rec\_tweet\_12"

val Tweet16 = "magic\_rec\_tweet\_16"

val Hashtag = "magic\_rec\_hashtag"

val UnreadBadgeCount17 = "magic\_rec\_unread\_badge\_count\_17"

val Highlights = "highlights"

val TweetAnalytics = "magic\_rec\_tweet\_analytics"

val Untracked = "untracked"

def crtToScribeTarget(crt: CommonRecommendationType): String = crt match {

case UserFollow =>

User2

case HermitUser =>

User4

case TweetRetweet | TweetFavorite =>

Tweet2

case TweetRetweetPhoto | TweetFavoritePhoto =>

Tweet4

case TweetRetweetVideo | TweetFavoriteVideo =>

Tweet5

case UrlTweetLanding =>

Tweet9

case F1FirstdegreeTweet | F1FirstdegreePhoto | F1FirstdegreeVideo =>

Tweet10

case AuthorTargetingTweet =>

Tweet11

case PeriscopeShare =>

Tweet12

case CommonRecommendationType.Highlights =>

Highlights

case HashtagTweet | HashtagTweetRetweet =>

Hashtag

case PinnedTweet =>

Tweet16

case UnreadBadgeCount =>

UnreadBadgeCount17

case TweetImpressions =>

TweetAnalytics

case \_ =>

Untracked

}

}