package com.twitter.follow\_recommendations.common.models

import com.twitter.adserver.thriftscala.{DisplayLocation => AdDisplayLocation}

import com.twitter.follow\_recommendations.logging.thriftscala.{

OfflineDisplayLocation => TOfflineDisplayLocation

}

import com.twitter.follow\_recommendations.thriftscala.{DisplayLocation => TDisplayLocation}

sealed trait DisplayLocation {

def toThrift: TDisplayLocation

def toOfflineThrift: TOfflineDisplayLocation

def toFsName: String

// corresponding display location in adserver if available

// make sure to be consistent with the definition here

def toAdDisplayLocation: Option[AdDisplayLocation] = None

}

/\*\*

\* Make sure you add the new DL to the following files and redeploy our attribution jobs

\* - follow-recommendations-service/thrift/src/main/thrift/display\_location.thrift

\* - follow-recommendations-service/thrift/src/main/thrift/logging/display\_location.thrift

\* - follow-recommendations-service/common/src/main/scala/com/twitter/follow\_recommendations/common/models/DisplayLocation.scala

\*/

object DisplayLocation {

case object ProfileSidebar extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.ProfileSidebar

override val toOfflineThrift: TOfflineDisplayLocation = TOfflineDisplayLocation.ProfileSidebar

override val toFsName: String = "ProfileSidebar"

override val toAdDisplayLocation: Option[AdDisplayLocation] = Some(

AdDisplayLocation.ProfileAccountsSidebar

)

}

case object HomeTimeline extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.HomeTimeline

override val toOfflineThrift: TOfflineDisplayLocation = TOfflineDisplayLocation.HomeTimeline

override val toFsName: String = "HomeTimeline"

override val toAdDisplayLocation: Option[AdDisplayLocation] = Some(

// it is based on the logic that HTL DL should correspond to Sidebar:

AdDisplayLocation.WtfSidebar

)

}

case object ReactiveFollow extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.ReactiveFollow

override val toOfflineThrift: TOfflineDisplayLocation = TOfflineDisplayLocation.ReactiveFollow

override val toFsName: String = "ReactiveFollow"

}

case object ExploreTab extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.ExploreTab

override val toOfflineThrift: TOfflineDisplayLocation = TOfflineDisplayLocation.ExploreTab

override val toFsName: String = "ExploreTab"

}

case object MagicRecs extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.MagicRecs

override val toOfflineThrift: TOfflineDisplayLocation = TOfflineDisplayLocation.MagicRecs

override val toFsName: String = "MagicRecs"

}

case object AbUploadInjection extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.AbUploadInjection

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.AbUploadInjection

override val toFsName: String = "AbUploadInjection"

}

case object RuxLandingPage extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.RuxLandingPage

override val toOfflineThrift: TOfflineDisplayLocation = TOfflineDisplayLocation.RuxLandingPage

override val toFsName: String = "RuxLandingPage"

}

case object ProfileBonusFollow extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.ProfileBonusFollow

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.ProfileBonusFollow

override val toFsName: String = "ProfileBonusFollow"

}

case object ElectionExploreWtf extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.ElectionExploreWtf

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.ElectionExploreWtf

override val toFsName: String = "ElectionExploreWtf"

}

case object ClusterFollow extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.ClusterFollow

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.ClusterFollow

override val toFsName: String = "ClusterFollow"

override val toAdDisplayLocation: Option[AdDisplayLocation] = Some(

AdDisplayLocation.ClusterFollow

)

}

case object HtlBonusFollow extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.HtlBonusFollow

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.HtlBonusFollow

override val toFsName: String = "HtlBonusFollow"

}

case object TopicLandingPageHeader extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.TopicLandingPageHeader

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.TopicLandingPageHeader

override val toFsName: String = "TopicLandingPageHeader"

}

case object NewUserSarusBackfill extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.NewUserSarusBackfill

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.NewUserSarusBackfill

override val toFsName: String = "NewUserSarusBackfill"

}

case object NuxPymk extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.NuxPymk

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.NuxPymk

override val toFsName: String = "NuxPymk"

}

case object NuxInterests extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.NuxInterests

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.NuxInterests

override val toFsName: String = "NuxInterests"

}

case object NuxTopicBonusFollow extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.NuxTopicBonusFollow

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.NuxTopicBonusFollow

override val toFsName: String = "NuxTopicBonusFollow"

}

case object Sidebar extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.Sidebar

override val toOfflineThrift: TOfflineDisplayLocation = TOfflineDisplayLocation.Sidebar

override val toFsName: String = "Sidebar"

override val toAdDisplayLocation: Option[AdDisplayLocation] = Some(

AdDisplayLocation.WtfSidebar

)

}

case object CampaignForm extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.CampaignForm

override val toOfflineThrift: TOfflineDisplayLocation = TOfflineDisplayLocation.CampaignForm

override val toFsName: String = "CampaignForm"

}

case object ProfileTopFollowers extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.ProfileTopFollowers

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.ProfileTopFollowers

override val toFsName: String = "ProfileTopFollowers"

}

case object ProfileTopFollowing extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.ProfileTopFollowing

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.ProfileTopFollowing

override val toFsName: String = "ProfileTopFollowing"

}

case object RuxPymk extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.RuxPymk

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.RuxPymk

override val toFsName: String = "RuxPymk"

}

case object IndiaCovid19CuratedAccountsWtf extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.IndiaCovid19CuratedAccountsWtf

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.IndiaCovid19CuratedAccountsWtf

override val toFsName: String = "IndiaCovid19CuratedAccountsWtf"

}

case object PeoplePlusPlus extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.PeoplePlusPlus

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.PeoplePlusPlus

override val toFsName: String = "PeoplePlusPlus"

}

case object TweetNotificationRecs extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.TweetNotificationRecs

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.TweetNotificationRecs

override val toFsName: String = "TweetNotificationRecs"

}

case object ProfileDeviceFollow extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.ProfileDeviceFollow

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.ProfileDeviceFollow

override val toFsName: String = "ProfileDeviceFollow"

}

case object RecosBackfill extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.RecosBackfill

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.RecosBackfill

override val toFsName: String = "RecosBackfill"

}

case object HtlSpaceHosts extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.HtlSpaceHosts

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.HtlSpaceHosts

override val toFsName: String = "HtlSpaceHosts"

}

case object PostNuxFollowTask extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.PostNuxFollowTask

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.PostNuxFollowTask

override val toFsName: String = "PostNuxFollowTask"

}

case object TopicLandingPage extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.TopicLandingPage

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.TopicLandingPage

override val toFsName: String = "TopicLandingPage"

}

case object UserTypeaheadPrefetch extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.UserTypeaheadPrefetch

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.UserTypeaheadPrefetch

override val toFsName: String = "UserTypeaheadPrefetch"

}

case object HomeTimelineRelatableAccounts extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.HomeTimelineRelatableAccounts

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.HomeTimelineRelatableAccounts

override val toFsName: String = "HomeTimelineRelatableAccounts"

}

case object NuxGeoCategory extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.NuxGeoCategory

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.NuxGeoCategory

override val toFsName: String = "NuxGeoCategory"

}

case object NuxInterestsCategory extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.NuxInterestsCategory

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.NuxInterestsCategory

override val toFsName: String = "NuxInterestsCategory"

}

case object TopArticles extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.TopArticles

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.TopArticles

override val toFsName: String = "TopArticles"

}

case object NuxPymkCategory extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.NuxPymkCategory

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.NuxPymkCategory

override val toFsName: String = "NuxPymkCategory"

}

case object HomeTimelineTweetRecs extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.HomeTimelineTweetRecs

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.HomeTimelineTweetRecs

override val toFsName: String = "HomeTimelineTweetRecs"

}

case object HtlBulkFriendFollows extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.HtlBulkFriendFollows

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.HtlBulkFriendFollows

override val toFsName: String = "HtlBulkFriendFollows"

}

case object NuxAutoFollow extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.NuxAutoFollow

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.NuxAutoFollow

override val toFsName: String = "NuxAutoFollow"

}

case object SearchBonusFollow extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.SearchBonusFollow

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.SearchBonusFollow

override val toFsName: String = "SearchBonusFollow"

}

case object ContentRecommender extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.ContentRecommender

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.ContentRecommender

override val toFsName: String = "ContentRecommender"

}

case object HomeTimelineReverseChron extends DisplayLocation {

override val toThrift: TDisplayLocation = TDisplayLocation.HomeTimelineReverseChron

override val toOfflineThrift: TOfflineDisplayLocation =

TOfflineDisplayLocation.HomeTimelineReverseChron

override val toFsName: String = "HomeTimelineReverseChron"

}

def fromThrift(displayLocation: TDisplayLocation): DisplayLocation = displayLocation match {

case TDisplayLocation.ProfileSidebar => ProfileSidebar

case TDisplayLocation.HomeTimeline => HomeTimeline

case TDisplayLocation.MagicRecs => MagicRecs

case TDisplayLocation.AbUploadInjection => AbUploadInjection

case TDisplayLocation.RuxLandingPage => RuxLandingPage

case TDisplayLocation.ProfileBonusFollow => ProfileBonusFollow

case TDisplayLocation.ElectionExploreWtf => ElectionExploreWtf

case TDisplayLocation.ClusterFollow => ClusterFollow

case TDisplayLocation.HtlBonusFollow => HtlBonusFollow

case TDisplayLocation.ReactiveFollow => ReactiveFollow

case TDisplayLocation.TopicLandingPageHeader => TopicLandingPageHeader

case TDisplayLocation.NewUserSarusBackfill => NewUserSarusBackfill

case TDisplayLocation.NuxPymk => NuxPymk

case TDisplayLocation.NuxInterests => NuxInterests

case TDisplayLocation.NuxTopicBonusFollow => NuxTopicBonusFollow

case TDisplayLocation.ExploreTab => ExploreTab

case TDisplayLocation.Sidebar => Sidebar

case TDisplayLocation.CampaignForm => CampaignForm

case TDisplayLocation.ProfileTopFollowers => ProfileTopFollowers

case TDisplayLocation.ProfileTopFollowing => ProfileTopFollowing

case TDisplayLocation.RuxPymk => RuxPymk

case TDisplayLocation.IndiaCovid19CuratedAccountsWtf => IndiaCovid19CuratedAccountsWtf

case TDisplayLocation.PeoplePlusPlus => PeoplePlusPlus

case TDisplayLocation.TweetNotificationRecs => TweetNotificationRecs

case TDisplayLocation.ProfileDeviceFollow => ProfileDeviceFollow

case TDisplayLocation.RecosBackfill => RecosBackfill

case TDisplayLocation.HtlSpaceHosts => HtlSpaceHosts

case TDisplayLocation.PostNuxFollowTask => PostNuxFollowTask

case TDisplayLocation.TopicLandingPage => TopicLandingPage

case TDisplayLocation.UserTypeaheadPrefetch => UserTypeaheadPrefetch

case TDisplayLocation.HomeTimelineRelatableAccounts => HomeTimelineRelatableAccounts

case TDisplayLocation.NuxGeoCategory => NuxGeoCategory

case TDisplayLocation.NuxInterestsCategory => NuxInterestsCategory

case TDisplayLocation.TopArticles => TopArticles

case TDisplayLocation.NuxPymkCategory => NuxPymkCategory

case TDisplayLocation.HomeTimelineTweetRecs => HomeTimelineTweetRecs

case TDisplayLocation.HtlBulkFriendFollows => HtlBulkFriendFollows

case TDisplayLocation.NuxAutoFollow => NuxAutoFollow

case TDisplayLocation.SearchBonusFollow => SearchBonusFollow

case TDisplayLocation.ContentRecommender => ContentRecommender

case TDisplayLocation.HomeTimelineReverseChron => HomeTimelineReverseChron

case TDisplayLocation.EnumUnknownDisplayLocation(i) =>

throw new UnknownDisplayLocationException(

s"Unknown display location thrift enum with value: ${i}")

}

def fromOfflineThrift(displayLocation: TOfflineDisplayLocation): DisplayLocation =

displayLocation match {

case TOfflineDisplayLocation.ProfileSidebar => ProfileSidebar

case TOfflineDisplayLocation.HomeTimeline => HomeTimeline

case TOfflineDisplayLocation.MagicRecs => MagicRecs

case TOfflineDisplayLocation.AbUploadInjection => AbUploadInjection

case TOfflineDisplayLocation.RuxLandingPage => RuxLandingPage

case TOfflineDisplayLocation.ProfileBonusFollow => ProfileBonusFollow

case TOfflineDisplayLocation.ElectionExploreWtf => ElectionExploreWtf

case TOfflineDisplayLocation.ClusterFollow => ClusterFollow

case TOfflineDisplayLocation.HtlBonusFollow => HtlBonusFollow

case TOfflineDisplayLocation.TopicLandingPageHeader => TopicLandingPageHeader

case TOfflineDisplayLocation.NewUserSarusBackfill => NewUserSarusBackfill

case TOfflineDisplayLocation.NuxPymk => NuxPymk

case TOfflineDisplayLocation.NuxInterests => NuxInterests

case TOfflineDisplayLocation.NuxTopicBonusFollow => NuxTopicBonusFollow

case TOfflineDisplayLocation.ExploreTab => ExploreTab

case TOfflineDisplayLocation.ReactiveFollow => ReactiveFollow

case TOfflineDisplayLocation.Sidebar => Sidebar

case TOfflineDisplayLocation.CampaignForm => CampaignForm

case TOfflineDisplayLocation.ProfileTopFollowers => ProfileTopFollowers

case TOfflineDisplayLocation.ProfileTopFollowing => ProfileTopFollowing

case TOfflineDisplayLocation.RuxPymk => RuxPymk

case TOfflineDisplayLocation.IndiaCovid19CuratedAccountsWtf => IndiaCovid19CuratedAccountsWtf

case TOfflineDisplayLocation.PeoplePlusPlus => PeoplePlusPlus

case TOfflineDisplayLocation.TweetNotificationRecs => TweetNotificationRecs

case TOfflineDisplayLocation.ProfileDeviceFollow => ProfileDeviceFollow

case TOfflineDisplayLocation.RecosBackfill => RecosBackfill

case TOfflineDisplayLocation.HtlSpaceHosts => HtlSpaceHosts

case TOfflineDisplayLocation.PostNuxFollowTask => PostNuxFollowTask

case TOfflineDisplayLocation.TopicLandingPage => TopicLandingPage

case TOfflineDisplayLocation.UserTypeaheadPrefetch => UserTypeaheadPrefetch

case TOfflineDisplayLocation.HomeTimelineRelatableAccounts => HomeTimelineRelatableAccounts

case TOfflineDisplayLocation.NuxGeoCategory => NuxGeoCategory

case TOfflineDisplayLocation.NuxInterestsCategory => NuxInterestsCategory

case TOfflineDisplayLocation.TopArticles => TopArticles

case TOfflineDisplayLocation.NuxPymkCategory => NuxPymkCategory

case TOfflineDisplayLocation.HomeTimelineTweetRecs => HomeTimelineTweetRecs

case TOfflineDisplayLocation.HtlBulkFriendFollows => HtlBulkFriendFollows

case TOfflineDisplayLocation.NuxAutoFollow => NuxAutoFollow

case TOfflineDisplayLocation.SearchBonusFollow => SearchBonusFollow

case TOfflineDisplayLocation.ContentRecommender => ContentRecommender

case TOfflineDisplayLocation.HomeTimelineReverseChron => HomeTimelineReverseChron

case TOfflineDisplayLocation.EnumUnknownOfflineDisplayLocation(i) =>

throw new UnknownDisplayLocationException(

s"Unknown offline display location thrift enum with value: ${i}")

}

}

class UnknownDisplayLocationException(message: String) extends Exception(message)