include "flows.thrift"

include "recently\_engaged\_user\_id.thrift"

namespace java com.twitter.follow\_recommendations.thriftjava

#@namespace scala com.twitter.follow\_recommendations.thriftscala

#@namespace strato com.twitter.follow\_recommendations

struct Profile {

1: required i64 profileId(personalDataType='UserId')

}(hasPersonalData='true')

struct Search {

1: required string searchQuery(personalDataType='SearchQuery')

}(hasPersonalData='true')

struct Rux {

1: required i64 focalAuthorId(personalDataType='UserId')

}(hasPersonalData='true')

struct Topic {

1: required i64 topicId(personalDataType = 'TopicFollow')

}(hasPersonalData='true')

struct ReactiveFollow {

1: required list<i64> followedUserIds(personalDataType='UserId')

}(hasPersonalData='true')

struct NuxInterests {

1: optional flows.FlowContext flowContext // set for recommendation inside an interactive flow

2: optional list<i64> uttInterestIds // if provided, we use these interestIds for generating candidates instead of for example fetching user selected interests

}(hasPersonalData='true')

struct AdCampaignTarget {

1: required list<i64> similarToUserIds(personalDataType='UserId')

}(hasPersonalData='true')

struct ConnectTab {

1: required list<i64> byfSeedUserIds(personalDataType='UserId')

2: required list<i64> similarToUserIds(personalDataType='UserId')

3: required list<recently\_engaged\_user\_id.RecentlyEngagedUserId> recentlyEngagedUserIds

}(hasPersonalData='true')

struct SimilarToUser {

1: required i64 similarToUserId(personalDataType='UserId')

}(hasPersonalData='true')

struct PostNuxFollowTask {

1: optional flows.FlowContext flowContext // set for recommendation inside an interactive flow

}(hasPersonalData='true')

union DisplayContext {

1: Profile profile

2: Search search

3: Rux rux

4: Topic topic

5: ReactiveFollow reactiveFollow

6: NuxInterests nuxInterests

7: AdCampaignTarget adCampaignTarget

8: ConnectTab connectTab

9: SimilarToUser similarToUser

10: PostNuxFollowTask postNuxFollowTask

}(hasPersonalData='true')