namespace java com.twitter.cr\_mixer.thriftjava

#@namespace scala com.twitter.cr\_mixer.thriftscala

#@namespace strato com.twitter.cr\_mixer

include "product.thrift"

include "product\_context.thrift"

include "com/twitter/product\_mixer/core/client\_context.thrift"

struct FrsTweetRequest {

1: required client\_context.ClientContext clientContext

2: required product.Product product

3: optional product\_context.ProductContext productContext

# excludedUserIds - user ids to be excluded from FRS candidate generation

4: optional list<i64> excludedUserIds (personalDataType = 'UserId')

# excludedTweetIds - tweet ids to be excluded from Earlybird candidate generation

5: optional list<i64> excludedTweetIds (personalDataType = 'TweetId')

} (persisted='true', hasPersonalData='true')

struct FrsTweet {

1: required i64 tweetId (personalDataType = 'TweetId')

2: required i64 authorId (personalDataType = 'UserId')

# skip 3 in case we need tweet score in the future

# frsPrimarySource - which FRS candidate source is the primary one to generate this author

4: optional i32 frsPrimarySource

# frsCandidateSourceScores - FRS candidate sources and the scores for this author

# for i32 to algorithm mapping, see https://sourcegraph.twitter.biz/git.twitter.biz/source/-/blob/hermit/hermit-core/src/main/scala/com/twitter/hermit/constants/AlgorithmFeedbackTokens.scala?L12

5: optional map<i32, double> frsCandidateSourceScores

# frsPrimaryScore - the score of the FRS primary candidate source

6: optional double frsAuthorScore

} (persisted='true', hasPersonalData = 'true')

struct FrsTweetResponse {

1: required list<FrsTweet> tweets

} (persisted='true')