package com.twitter.recosinjector.clients

import com.twitter.finagle.stats.StatsReceiver

import com.twitter.stitch.tweetypie.TweetyPie.{TweetyPieException, TweetyPieResult}

import com.twitter.storehaus.ReadableStore

import com.twitter.tweetypie.thriftscala.Tweet

import com.twitter.util.Future

class Tweetypie(

tweetyPieStore: ReadableStore[Long, TweetyPieResult]

)(

implicit statsReceiver: StatsReceiver) {

private val stats = statsReceiver.scope(this.getClass.getSimpleName)

private val failureStats = stats.scope("getTweetFailure")

def getTweet(tweetId: Long): Future[Option[Tweet]] = {

tweetyPieStore

.get(tweetId)

.map { \_.map(\_.tweet) }

.rescue {

case e: TweetyPieException =>

// Usually results from trying to query a protected or unsafe tweet

failureStats.scope("TweetyPieException").counter(e.result.tweetState.toString).incr()

Future.None

case e =>

failureStats.counter(e.getClass.getSimpleName).incr()

Future.None

}

}

}