package com.twitter.tweetypie.repository

import com.twitter.stitch.Stitch

import com.twitter.strato.client.Fetcher

import com.twitter.strato.client.{Client => StratoClient}

import com.twitter.tweetypie.thriftscala.Tweet

import com.twitter.strato.thrift.ScroogeConvImplicits.\_

import com.twitter.vibes.thriftscala.VibeV2

object VibeRepository {

type Type = Tweet => Stitch[Option[VibeV2]]

val column = "vibes/vibe.Tweet"

case class VibeView(viewerId: Option[Long])

/\*\*

\* Creates a function that applies the vibes/vibe.Tweet strato column fetch on the given

\* Tweet. Strato column source: go/vibe.strato

\* @param client Strato client

\* @return

\*/

def apply(client: StratoClient): Type = {

val fetcher: Fetcher[Long, VibeView, VibeV2] =

client.fetcher[Long, VibeView, VibeV2](column)

tweet =>

fetcher

.fetch(tweet.id, VibeView(None))

.map(\_.v)

}

}