package com.twitter.timelineranker.util

import com.twitter.finagle.util.DefaultTimer

import com.twitter.servo.repository.Repository

import com.twitter.util.Duration

import com.twitter.util.Future

import com.twitter.util.Timer

import scala.util.Random

// Inject an artificial delay into an underlying repository's response to match the provided p50

// and max latencies.

class LatentRepository[Q, R](

underlying: Repository[Q, R],

p50: Duration,

max: Duration,

random: Random = new Random,

timer: Timer = DefaultTimer)

extends Repository[Q, R] {

import scala.math.ceil

import scala.math.pow

val p50Millis: Long = p50.inMilliseconds

val maxMillis: Long = max.inMilliseconds

require(p50Millis > 0 && maxMillis > 0 && maxMillis > p50Millis)

override def apply(query: Q): Future[R] = {

val x = random.nextDouble()

val sleepTime = ceil(pow(p50Millis, 2 \* (1 - x)) / pow(maxMillis, 1 - 2 \* x)).toInt

Future.sleep(Duration.fromMilliseconds(sleepTime))(timer).flatMap { \_ => underlying(query) }

}

}