package com.twitter.home\_mixer.module

import com.google.inject.Provides

import com.twitter.conversions.DurationOps.\_

import com.twitter.finagle.mtls.authentication.ServiceIdentifier

import com.twitter.home\_mixer.param.HomeMixerInjectionNames.RealGraphManhattanEndpoint

import com.twitter.inject.TwitterModule

import com.twitter.inject.annotations.Flag

import com.twitter.storage.client.manhattan.kv.\_

import com.twitter.timelines.config.ConfigUtils

import com.twitter.util.Duration

import javax.inject.Named

import javax.inject.Singleton

object ManhattanClientsModule extends TwitterModule with ConfigUtils {

private val ApolloDest = "/s/manhattan/apollo.native-thrift"

private final val Timeout = "mh\_real\_graph.timeout"

flag[Duration](Timeout, 150.millis, "Timeout total")

@Provides

@Singleton

@Named(RealGraphManhattanEndpoint)

def providesRealGraphManhattanEndpoint(

@Flag(Timeout) timeout: Duration,

serviceIdentifier: ServiceIdentifier

): ManhattanKVEndpoint = {

lazy val client = ManhattanKVClient(

appId = "real\_graph",

dest = ApolloDest,

mtlsParams = ManhattanKVClientMtlsParams(serviceIdentifier = serviceIdentifier),

label = "real-graph-data"

)

ManhattanKVEndpointBuilder(client)

.maxRetryCount(2)

.defaultMaxTimeout(timeout)

.build()

}

}