package com.twitter.servo.util

import com.twitter.finagle.stats.\_

/\*\*

\* Stores scoped StatsReceivers in a map to avoid unnecessary object creation.

\*/

class MemoizingStatsReceiver(val self: StatsReceiver)

extends StatsReceiver

with DelegatingStatsReceiver

with Proxy {

def underlying: Seq[StatsReceiver] = Seq(self)

val repr = self.repr

private[this] lazy val scopeMemo =

Memoize[String, StatsReceiver] { name =>

new MemoizingStatsReceiver(self.scope(name))

}

private[this] lazy val counterMemo =

Memoize[(Seq[String], Verbosity), Counter] {

case (names, verbosity) =>

self.counter(verbosity, names: \_\*)

}

private[this] lazy val statMemo =

Memoize[(Seq[String], Verbosity), Stat] {

case (names, verbosity) =>

self.stat(verbosity, names: \_\*)

}

def counter(metricBuilder: MetricBuilder): Counter =

counterMemo(metricBuilder.name -> metricBuilder.verbosity)

def stat(metricBuilder: MetricBuilder): Stat = statMemo(

metricBuilder.name -> metricBuilder.verbosity)

def addGauge(metricBuilder: MetricBuilder)(f: => Float): Gauge = {

// scalafix:off StoreGaugesAsMemberVariables

self.addGauge(metricBuilder)(f)

// scalafix:on StoreGaugesAsMemberVariables

}

override def scope(name: String): StatsReceiver = scopeMemo(name)

}