package com.twitter.visibility.interfaces.cards

import com.twitter.finagle.stats.StatsReceiver

import com.twitter.stitch.Stitch

import com.twitter.visibility.builder.VisibilityResult

class CardVisibilityLibraryParityTest(statsReceiver: StatsReceiver) {

private val parityTestScope = statsReceiver.scope("card\_visibility\_library\_parity")

private val requests = parityTestScope.counter("requests")

private val equal = parityTestScope.counter("equal")

private val incorrect = parityTestScope.counter("incorrect")

private val failures = parityTestScope.counter("failures")

def runParityTest(

preHydratedFeatureVisibilityResult: Stitch[VisibilityResult],

resp: VisibilityResult

): Stitch[Unit] = {

requests.incr()

preHydratedFeatureVisibilityResult

.flatMap { parityResponse =>

if (parityResponse.verdict == resp.verdict) {

equal.incr()

} else {

incorrect.incr()

}

Stitch.Done

}.rescue {

case t: Throwable =>

failures.incr()

Stitch.Done

}.unit

}

}