package com.twitter.servo.util

import com.twitter.util.Throwables

/\*\*

\* An object with some helper methods for dealing with exceptions

\* (currently just classname cleanup)

\*/

object ThrowableHelper {

/\*\*

\* Returns a sanitized sequence of classname for the given Throwable

\* including root causes.

\*/

def sanitizeClassnameChain(t: Throwable): Seq[String] =

Throwables.mkString(t).map(classnameTransform(\_))

/\*\*

\* Returns a sanitized classname for the given Throwable.

\*/

def sanitizeClassname(t: Throwable): String =

classnameTransform(t.getClass.getName)

/\*\*

\* A function that applies a bunch of cleanup transformations to exception classnames

\* (currently just 1, but there will likely be more!).

\*/

private val classnameTransform: String => String =

Memoize { stripSuffix("$Immutable").andThen(stripSuffix("$")) }

/\*\*

\* Generates a function that strips off the specified suffix from strings, if found.

\*/

private def stripSuffix(suffix: String): String => String =

s => {

if (s.endsWith(suffix))

s.substring(0, s.length - suffix.length)

else

s

}

}