package com.twitter.tweetypie.util.logging

import ch.qos.logback.classic.Level

import ch.qos.logback.classic.spi.ILoggingEvent

import ch.qos.logback.core.filter.Filter

import ch.qos.logback.core.spi.FilterReply

/\*\*

\* This class is currently being used by logback to log statements from tweetypie at one level and

\* log statements from other packages at another.

\*

\* Filters do not change the log levels of individual loggers. Filters filter out specific messages

\* for specific appenders. This allows us to have a log file with lots of information you will

\* mostly not need and a log file with only important information. This type of filtering cannot be

\* accomplished by changing the log levels of loggers, because the logger levels are global. We want

\* to change the semantics for specific destinations (appenders).

\*/

class OnlyImportantLogsLoggingFilter extends Filter[ILoggingEvent] {

private[this] def notImportant(loggerName: String): Boolean =

!loggerName.startsWith("com.twitter.tweetypie")

override def decide(event: ILoggingEvent): FilterReply =

if (!isStarted || event.getLevel.isGreaterOrEqual(Level.WARN)) {

FilterReply.NEUTRAL

} else if (notImportant(event.getLoggerName())) {

FilterReply.DENY

} else {

FilterReply.NEUTRAL

}

}