<configuration>

<shutdownHook class="ch.qos.logback.core.hook.DelayingShutdownHook"/>

<property name="async\_queue\_size" value="${queue.size:-50000}"/>

<property name="async\_max\_flush\_time" value="${max.flush.time:-0}"/>

<!-- ===================================================== -->

<!-- Structured Logging -->

<!-- ===================================================== -->

<!-- Only sample 0.1% of the requests -->

<property name="splunk\_sampling\_rate" value="${splunk\_sampling\_rate:-0.001}"/>

<include resource="structured-logger-logback.xml"/>

<!-- ===================================================== -->

<!-- Service Config -->

<!-- ===================================================== -->

<property name="DEFAULT\_SERVICE\_PATTERN"

value="%-16X{transactionId} %logger %msg"/>

<!-- ===================================================== -->

<!-- Common Config -->

<!-- ===================================================== -->

<!-- JUL/JDK14 to Logback bridge -->

<contextListener class="ch.qos.logback.classic.jul.LevelChangePropagator">

<resetJUL>true</resetJUL>

</contextListener>

<!-- Service Log (Rollover every 50MB, max 11 logs) -->

<appender name="SERVICE" class="ch.qos.logback.core.rolling.RollingFileAppender">

<file>${log.service.output}</file>

<rollingPolicy class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">

<fileNamePattern>${log.service.output}.%i</fileNamePattern>

<minIndex>1</minIndex>

<maxIndex>10</maxIndex>

</rollingPolicy>

<triggeringPolicy class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">

<maxFileSize>50MB</maxFileSize>

</triggeringPolicy>

<encoder>

<pattern>%date %.-3level ${DEFAULT\_SERVICE\_PATTERN}%n</pattern>

</encoder>

</appender>

<!-- Strato package only log (Rollover every 50MB, max 11 logs) -->

<appender name="STRATO-ONLY" class="ch.qos.logback.core.rolling.RollingFileAppender">

<file>${log.strato\_only.output}</file>

<rollingPolicy class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">

<fileNamePattern>${log.strato\_only.output}.%i</fileNamePattern>

<minIndex>1</minIndex>

<maxIndex>10</maxIndex>

</rollingPolicy>

<triggeringPolicy class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">

<maxFileSize>50MB</maxFileSize>

</triggeringPolicy>

<encoder>

<pattern>%date %.-3level ${DEFAULT\_SERVICE\_PATTERN}%n</pattern>

</encoder>

</appender>

<!-- LogLens -->

<appender name="LOGLENS" class="com.twitter.loglens.logback.LoglensAppender">

<mdcAdditionalContext>true</mdcAdditionalContext>

<category>loglens</category>

<index>${log.lens.index}</index>

<tag>${log.lens.tag}/service</tag>

<encoder>

<pattern>%msg%n</pattern>

</encoder>

<turboFilter class="ch.qos.logback.classic.turbo.DuplicateMessageFilter">

<cacheSize>500</cacheSize>

<allowedRepetitions>50</allowedRepetitions>

</turboFilter>

<filter class="com.twitter.strato.logging.logback.RegexFilter">

<forLogger>manhattan-client</forLogger>

<excludeRegex>.\*InvalidRequest.\*</excludeRegex>

</filter>

</appender>

<!-- ===================================================== -->

<!-- Primary Async Appenders -->

<!-- ===================================================== -->

<appender name="ASYNC-SERVICE" class="ch.qos.logback.classic.AsyncAppender">

<queueSize>${async\_queue\_size}</queueSize>

<maxFlushTime>${async\_max\_flush\_time}</maxFlushTime>

<appender-ref ref="SERVICE"/>

</appender>

<appender name="ASYNC-STRATO-ONLY" class="ch.qos.logback.classic.AsyncAppender">

<queueSize>${async\_queue\_size}</queueSize>

<maxFlushTime>${async\_max\_flush\_time}</maxFlushTime>

<appender-ref ref="STRATO-ONLY"/>

</appender>

<appender name="ASYNC-LOGLENS" class="ch.qos.logback.classic.AsyncAppender">

<queueSize>${async\_queue\_size}</queueSize>

<maxFlushTime>${async\_max\_flush\_time}</maxFlushTime>

<appender-ref ref="LOGLENS"/>

</appender>

<!-- ===================================================== -->

<!-- Package Config -->

<!-- ===================================================== -->

<!-- Per-Package Config (shared) -->

<logger name="com.twitter" level="info"/>

<!--

By default, we leave the strato package at INFO level.

However, this line allows us to set the entire strato package, or a subset of it, to

a specific level. For example, if you pass -Dstrato\_log\_package=streaming -Dstrato\_log\_level=DEBUG

only loggers under com.twitter.strato.streaming.\* will be set to DEBUG level. Passing only

-Dstrato\_log\_level will set all of strato.\* to the specified level.

-->

<logger name="com.twitter.strato${strato\_log\_package:-}" level="${strato\_log\_level:-INFO}"/>

<logger name="com.twitter.wilyns" level="warn"/>

<logger name="com.twitter.finagle.mux" level="warn"/>

<logger name="com.twitter.finagle.serverset2" level="warn"/>

<logger name="com.twitter.logging.ScribeHandler" level="warn"/>

<logger name="com.twitter.zookeeper.client.internal" level="warn"/>

<logger name="com.twitter.decider.StoreDecider" level="warn"/>

<!-- Per-Package Config (Strato) -->

<logger name="com.twitter.distributedlog.client" level="warn"/>

<logger name="com.twitter.finagle.mtls.authorization.config.AccessControlListConfiguration" level="warn"/>

<logger name="com.twitter.finatra.kafka.common.kerberoshelpers" level="warn"/>

<logger name="com.twitter.finatra.kafka.utils.BootstrapServerUtils" level="warn"/>

<logger name="com.twitter.server.coordinate" level="error"/>

<logger name="com.twitter.zookeeper.client" level="info"/>

<logger name="org.apache.zookeeper" level="error"/>

<logger name="org.apache.zookeeper.ClientCnxn" level="warn"/>

<logger name="ZkSession" level="info"/>

<logger name="OptimisticLockingCache" level="off"/>

<logger name="manhattan-client" level="warn"/>

<logger name="strato.op" level="warn"/>

<logger name="org.apache.kafka.clients.NetworkClient" level="error"/>

<logger name="org.apache.kafka.clients.consumer.internals" level="error"/>

<logger name="org.apache.kafka.clients.producer.internals" level="error"/>

<!-- produce a lot of messages like: Building client authenticator with server name kafka -->

<logger name="org.apache.kafka.common.network" level="warn"/>

<!-- Root Config -->

<root level="${log\_level:-INFO}">

<appender-ref ref="ASYNC-SERVICE"/>

<appender-ref ref="ASYNC-LOGLENS"/>

</root>

<!-- Strato package only logging-->

<logger name="com.twitter.strato"

level="info"

additivity="true">

<appender-ref ref="ASYNC-STRATO-ONLY" />

</logger>

</configuration>