日志插件使用说明

准备

要使用日志插件，首先需要在工程中进行以下几项配置：

* **引入jar包**

1. 引入log插件包

|  |
| --- |
| <dependency>      <groupId>com.ruwe</groupId>      <artifactId>log</artifactId>      <version>1.0-SNAPSHOT</version>  </dependency> |

1. 引入修改后的dubbo包

|  |
| --- |
| <dependency>      <groupId>com.alibaba</groupId>      <artifactId>dubbo</artifactId>      <version>2.5.3</version>      <exclusions>          <exclusion>          <groupId>org.springframework</groupId>          <artifactId>spring</artifactId>          </exclusion>      </exclusions>  </dependency> |

* **配置logback**

复制配置文件到工程中，配置文件内容如下：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8"?>  <configuration scan="true" scanPeriod="3 seconds" debug="false">      <property name="AppName" value="LogCollector" />      <contextName>${AppName}</contextName>      <appender name="stdout" class="ch.qos.logback.core.ConsoleAppender">          <encoder>              <pattern> %d{yyyy-MM-dd HH:mm:ss.SSS} %c [%thread] %-5level %logger{36} - %msg%n</pattern>              <charset>UTF-8</charset>          </encoder>      </appender>      <appender name="ERROR\_APPENDER" class="ch.qos.logback.core.rolling.RollingFileAppender">          <file>${CATALINA\_BASE}/ruwe\_log/error/error.out</file>          <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">              <fileNamePattern>${CATALINA\_BASE}/ruwe\_log/error/error.%d{yyyy-MM-dd}.log</fileNamePattern>              <maxHistory>30</maxHistory>          </rollingPolicy>          <encoder charset="UTF-8">              <pattern>%d{yyyy-MM-dd HH:mm:ss.SSS} [%thread] %-5level %logger{36} - %msg%n</pattern>              <target>System.err</target>          </encoder>          <filter class="ch.qos.logback.classic.filter.LevelFilter">              <level>ERROR</level>              <onMatch>ACCEPT</onMatch>              <onMismatch>DENY</onMismatch>          </filter>      </appender>      <appender name="WEB\_APPENDER" class="ch.qos.logback.core.rolling.RollingFileAppender">          <file>${CATALINA\_BASE}/ruwe\_log/web/web.out</file>          <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">              <fileNamePattern>${CATALINA\_BASE}/ruwe\_log/web/web\_%d{yyyyMMddHH}.log</fileNamePattern>              <maxHistory>30</maxHistory>          </rollingPolicy>          <encoder>              <pattern>%d{yyyy-MM-dd HH:mm:ss.SSS} [%thread] %-5level %logger{36} - %msg%n</pattern>          </encoder>      </appender>      <appender name="COMMON\_APPENDER" class="ch.qos.logback.core.rolling.RollingFileAppender">          <file>${CATALINA\_BASE}/ruwe\_log/common/common.out</file>          <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">              <fileNamePattern>${CATALINA\_BASE}/ruwe\_log/common/common\_%d{yyyyMMddHH}.log</fileNamePattern>              <maxHistory>30</maxHistory>          </rollingPolicy>          <encoder>              <pattern>%d{yyyy-MM-dd HH:mm:ss.SSS} [%thread] %-5level %logger{36} - %msg%n</pattern>          </encoder>      </appender>      <appender name="ALL" class="ch.qos.logback.core.rolling.RollingFileAppender">          <file>${CATALINA\_BASE}/ruwe\_log/all/all.out</file>          <rollingPolicy class="ch.qos.logback.core.rolling.TimeBasedRollingPolicy">              <fileNamePattern>${CATALINA\_BASE}/ruwe\_log/all/all.%d{yyyy-MM-dd}.log</fileNamePattern>              <maxHistory>30</maxHistory>          </rollingPolicy>          <encoder charset="UTF-8">              <pattern>%d{yyyy-MM-dd HH:mm:ss.SSS} [%thread] %-5level %logger{36} - %msg%n</pattern>          </encoder>      </appender>      <root level="DEBUG">          <appender-ref ref="stdout" />          <appender-ref ref="ALL"/>          <appender-ref ref="ERROR\_APPENDER"/>          <appender-ref ref="WEB\_APPENDER"/>          <appender-ref ref="COMMON\_APPENDER"/>      </root>      <!--能用日志配置-->      <logger name="WEB" level="INFO" additivity="false">          <appender-ref ref="stdout"/>          <appender-ref ref="WEB\_APPENDER"/>      </logger>      <logger name="COMMON" level="INFO" additivity="false">          <appender-ref ref="stdout"/>          <appender-ref ref="COMMON\_APPENDER"/>      </logger>      <logger name="org.springframework" level="WARN">      </logger>      <logger name="org.apache" level="WARN">      </logger>  </configuration> |

* **声明AOP拦截**

日志AOP拦截器分为两种，一种拦截Web层日志(即HttpServlet级别的)，另外一种拦截业务处理层的日志。日志拦截器配置在Spring配置文件中。

1. web层拦截

|  |
| --- |
| <bean id="webLogger" class="com.ruwe.log.aspect.WebLogger" >      <!--不能为空 注意需要根据所在的微服务配置微服务名称，具体微服务名称查看log jar中的MSName.properties -->      <property name="msName" value="KOULIANG\_API" />      <!--不能为空 注意根据日志类型设置该值，web层目前有API和THIRD，可以查看log jar中的LogType.java -->      <property name="logType" value="API" />  </bean>    <aop:config>      <!-- aop顺序:放在业务aop之后 -->      <aop:aspect id="webAspect" ref="webLogger" order="100">          <!-- 所有web层都通过表达式RequestMapping进行匹配-->          <aop:pointcut id="webLog" expression="@annotation(org.springframework.web.bind.annotation.RequestMapping)"/>          <aop:around method="record" pointcut-ref="webLog"/>       </aop:aspect>  </aop:config> |

1. 业务层拦截

|  |
| --- |
| <bean id="commonLogger" class="com.ruwe.log.aspect.CommonLogger" >      <!--不能为空 注意需要根据所在的微服务配置微服务名称，具体微服务名称查看log jar中的MSName.properties -->      <property name="msName" value="USER\_CENTER" />      <!--不能为空 注意根据日志类型设置该值，业务层目前有DB、PRO、JOB，可以查看log jar中的LogType.java -->      <property name="logType" value="PRO" />  </bean>  <aop:config>      <aop:aspect id="commonLogAspect" ref="commonLogger">          <!-- 根据不同的业务匹配包 -->          <aop:pointcut id="commonLog" expression="execution(\* com.ruwe.business.service.\*.\*(..))"/>          <aop:around method="record" pointcut-ref="commonLog"/>      </aop:aspect>  </aop:config> |

使用

1. 声明Logger

|  |
| --- |
| private static final Logger LOGGER = LoggerFactory.getLogger(HelloController.class); |

1. 使用

|  |
| --- |
| LogUtils.debug(LOGGER, "title", "key", "value");  LogUtils.info(LOGGER, "info", "msg", user);  LogUtils.error(LOGGER, "error", "1", user); |

1. LogUtils包含debug、info、warn、error等方法，参数和之前的LogUtils一样，只是这个LogUtils是log jar中 com.ruwe.log.util 包里的LogUtils.java。