

Logging

- [Current state](#)
 - [Important note](#)
 - [Tika parser modules](#)
- [Example configuration for Apache Tika 2.5.0+ \(Apache Maven\)](#)
 - [Common sections](#)
 - [Apache Log4j 2.x with slf4j bridges](#)
 - [Logback](#)
 - [Apache Log4j 2.x with native bridges](#)
- [Example configuration for Apache Tika 2.5.0+ \(Gradle\)](#)
 - [Common sections](#)
 - [Apache Log4j 2.x with slf4j bridges](#)
 - [Logback](#)
 - [Apache Log4j 2.x with native bridges](#)

Current state

Apache Tika include a lot of Apache and thirdparty libraries that have different approach to logging. Tika use `slf4j-api` as logging API and [Apache Log4j 2.x](#) as an implementation for modules that require it.

Important note

Since Tika 2.5.0 (released 2022-10-03) depends on `slf4j-api` 2.0.x which requires downstream library users to update logging backend to compatible version. Tika 2.0.0 – 2.4.1 depends on `slf4j-api` 1.7.x.

Otherwise you will receive something like following message:

```
SLF4J: No SLF4J providers were found.
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See https://www.slf4j.org/codes.html#noProviders for further details.
SLF4J: Class path contains SLF4J bindings targeting slf4j-api versions 1.7.x or earlier.
SLF4J: Ignoring binding found at file:/home/gross/.gradle/caches/modules-2/files-2.1/org.jboss.slf4j/slf4j-jboss-logmanager/1.2.0.Final/baff8ae78011e6859e127a5cb6f16332a056fd93/slf4j-jboss-logmanager-1.2.0.Final.jar!/org/slf4j/impl/StaticLoggerBinder.class
SLF4J: See https://www.slf4j.org/codes.html#ignoredBindings for an explanation.
ERROR StatusLogger Log4j2 could not find a logging implementation. Please add log4j-core to the classpath. Using SimpleLogger to log to the console..
```

Updates for popular logging backends:

- [Apache Log4j 2.x](#): `org.apache.logging.log4j:log4j-slf4j-impl` `org.apache.logging.log4j:log4j-slf4j2-impl`
- [Logback](#): `ch.qos.logback:logback-classic` 1.2.x 1.3.x (uses older `javax.*` APIs) or 1.4.x (uses `jakarta.*` APIs)
- [Apache Log4j 1.2.x](#): `org.slf4j:slf4j-log4j12` `org.slf4j:slf4j-reload4j` (though `slf4j-log4j12` has relocation relocation directive to `slf4j-reload4j` since 1.7.34) or migrate to the [Log4j 2.x](#) since `log4j 1.2.x` is in the End of Life status since 2015 and has [known vulnerabilities](#)

JBoss Logging (`slf4j-jboss-logging/slf4j-jboss-logmanager`) as of 2022-11-09 are still on `slf4j-api` 1.7.x, see <https://issues.redhat.com/browse/JBLOGGING-165>. Currently you can try downgrading `org.slf4j:slf4j-api` version to 1.7.36 if you have to use Tika with JBoss Logging (e.g. if you use Quarkus or WildFly native logging).

Tika parser modules

`tika-parser-*-module` artifacts depend on many Apache and thirdparty libraries. Tika itself use `slf4j-api` but underlying libraries use different logging API ([commons-logging](#), [java.util.logging](#), [log4j 1.2.x](#), [log4j 2.x](#), [slf4j](#)).

By default Tika will bring `slf4j-api` via `tika-core` and some bridges like `org.slf4j:jcl-over-slf4j` and `org.slf4j:jul-to-slf4j` as opinionated default. Depending on your logging backend and preferred configuration you'll need different dependency exclusions and bridges /implementations.

In you have no preference about logging backend it's enough to add `org.apache.logging.log4j:log4j-core`, `org.apache.logging.log4j:log4j-slf4j2-impl` and `org.apache.logging.log4j:log4j-1.2-api` (or `org.slf4j:log4j-over-slf4j`) and exclude `log4j:log4j`, `commons-logging:commons-logging`, `ch.qos.logback:logback-classic`, `ch.qos.logback:logback-core`, `ch.qos.reload4j:reload4j` and `org.slf4j:slf4j-reload4j`.

As of `main` branch (and Tika 2.6.0) all Tika source use `slf4j-api` as a logging API with `org.apache.logging.log4j:log4j-core:2.x` as the backend for applications like `tika-app` / `tika-eval-app` / `tika-server`.

Following sections shows how to configure different logging solutions/backends dependencies to avoid conflicts. Loggers configuration are out of scope of this document, you should look at relevant library documentation.

Example configuration for Apache Tika 2.5.0+ (Apache Maven)

If you use [Apache Maven](#) dependency section in `pom.xml` will contain something like this:

Common sections

```
<!-- Merge with your properties section -->
<properties>
  <!-- components versions, feel free keep only required for your case -->
  <tika.version>2.6.0</tika.version>
  <slf4j.version>2.0.3</slf4j.version>
  <log4j2.version>2.19.0</log4j2.version>
  <logback.version>1.4.4</logback.version> <!-- 1.4.4 for Jakarta EE 9+ or 1.3.4 if you use Java EE or Jakarta EE
8 -->
  <reload4j.version>1.2.22</reload4j.version>
</properties>

<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.apache.tika</groupId>
      <artifactId>tika-bom</artifactId>
      <version>${tika.version}</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
    <dependency>
      <groupId>org.apache.logging.log4j</groupId>
      <artifactId>log4j-bom</artifactId>
      <version>${log4j2.version}</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>

<!-- Merge with your dependencies section -->
<dependencies>
  <dependency>
    <groupId>org.apache.tika</groupId>
    <artifactId>tika-core</artifactId>
  </dependency>
  <dependency>
    <groupId>org.apache.tika</groupId>
    <artifactId>tika-parsers-standard-package</artifactId>
    <exclusions>
      <!--
      This exclusions will become obsolete at some point but better to keep it now.
      tika-parser-* module should exclude commons-logging explicitly but upstream libraries
      may add it to their transitive dependencies
      -->
      <exclusion>
        <groupId>commons-logging</groupId>
        <artifactId>commons-logging</artifactId>
      </exclusion>

      <!--
      These exclusions aren't necessary for tika-parsers-standard-package
      but may be required for other artifacts to have explicit logging configuration
      and avoid logging backend loops.
      -->
      <exclusion>
        <groupId>log4j</groupId>
        <artifactId>log4j</artifactId>
      </exclusion>
      <exclusion>
        <groupId>org.slf4j</groupId>
        <artifactId>slf4j-log4j12</artifactId>
      </exclusion>
      <exclusion>
        <groupId>org.slf4j</groupId>
        <artifactId>slf4j-reload4j</artifactId>
      </exclusion>
      <exclusion>
        <groupId>ch.qos.logback</groupId>
        <artifactId>logback-core</artifactId>
      </exclusion>
      <exclusion>
        <groupId>ch.qos.logback</groupId>
        <artifactId>logback-classic</artifactId>
      </exclusion>
    </exclusions>
  </dependency>
</dependencies>
```

```

        <exclusion>
            <groupId>ch.qos.reload4j</groupId>
            <artifactId>reload4j</artifactId>
        </exclusion>
    </exclusions>
</dependency>

<!-- You may want to add these dependencies to the dependencyManagement to force consistent versions and omit
their versions here -->
<dependency>
    <groupId>org.slf4j</groupId>
    <artifactId>slf4j-api</artifactId>
    <version>${slf4j.version}</version>
</dependency>

<!-- java.util.logging to slf4j adapter, requires additional configuration, see https://www.slf4j.org/api/org/slf4j/bridge/SLF4JBridgeHandler.html -->
<dependency>
    <groupId>org.slf4j</groupId>
    <artifactId>jul-to-slf4j</artifactId>
    <version>${slf4j.version}</version>
</dependency>

<!-- commons-logging (JCL) to slf4j bridge -->
<dependency>
    <groupId>org.slf4j</groupId>
    <artifactId>jcl-over-slf4j</artifactId>
    <version>${slf4j.version}</version>
    <scope>runtime</scope>
</dependency>

<!-- log4j 1.2.x to slf4j bridge -->
<dependency>
    <groupId>org.slf4j</groupId>
    <artifactId>log4j-over-slf4j</artifactId>
    <version>${slf4j.version}</version>
    <scope>runtime</scope>
</dependency>
</dependencies>

```

Apache Log4j 2.x with slf4j bridges

```

<!-- Merge with your dependencies section -->
<dependencies>
    <!-- logging backend: log4j 2.x -->
    <dependency>
        <groupId>org.apache.logging.log4j</groupId>
        <artifactId>log4j-api</artifactId>
        <!-- version is omitted since there's org.apache.logging.log4j:log4j-bom in the dependencyManagement section -->
    </dependency>
    <dependency>
        <groupId>org.apache.logging.log4j</groupId>
        <artifactId>log4j-core</artifactId>
        <!-- version is omitted since there's org.apache.logging.log4j:log4j-bom in the dependencyManagement section -->
        <scope>runtime</scope>
    </dependency>

    <!-- slf4j implementation that forwards to log4j 2.x -->
    <dependency>
        <groupId>org.apache.logging.log4j</groupId>
        <artifactId>log4j-slf4j2-impl</artifactId> <!-- for slf4j 1.7.x use log4j-slf4j-impl instead -->
        <!-- version is omitted since there's org.apache.logging.log4j:log4j-bom in the dependencyManagement section -->
        <scope>runtime</scope>
    </dependency>
</dependencies>

```

Logback

```

<!-- Merge with your dependencies section -->
<dependencies>
  <!-- slf4j implementation -->
  <dependency>
    <groupId>ch.qos.logback</groupId>
    <artifactId>logback-classic</artifactId>
    <version>${logback.version}</version>
    <scope>runtime</scope>
  </dependency>

  <!-- log4j2 to slf4j adapter -->
  <dependency>
    <groupId>org.apache.logging.log4j</groupId>
    <artifactId>log4j-to-slf4j</artifactId>
    <!-- version is omitted since there's org.apache.logging.log4j:log4j-bom in the dependencyManagement section -->
    <scope>runtime</scope>
  </dependency>
</dependencies>

```

Apache Log4j 2.x with native bridges

```

<dependencies>
  <dependency>
    <groupId>org.apache.tika</groupId>
    <artifactId>tika-parsers-standard-package</artifactId>
    <exclusions>
      <exclusion>
        <groupId>commons-logging</groupId>
        <artifactId>commons-logging</artifactId>
      </exclusion>
      <exclusion>
        <groupId>log4j</groupId>
        <artifactId>log4j</artifactId>
      </exclusion>
      <exclusion>
        <groupId>org.slf4j</groupId>
        <artifactId>slf4j-log4j12</artifactId>
      </exclusion>
      <exclusion>
        <groupId>org.slf4j</groupId>
        <artifactId>slf4j-reload4j</artifactId>
      </exclusion>
      <exclusion>
        <groupId>ch.qos.logback</groupId>
        <artifactId>logback-core</artifactId>
      </exclusion>
      <exclusion>
        <groupId>ch.qos.logback</groupId>
        <artifactId>logback-classic</artifactId>
      </exclusion>
      <exclusion>
        <groupId>ch.qos.reload4j</groupId>
        <artifactId>reload4j</artifactId>
      </exclusion>
    </exclusions>

    <!-- Additionally exclude slf4j bridges -->
    <exclusion>
      <groupId>org.slf4j</groupId>
      <artifactId>jul-to-slf4j</artifactId>
    </exclusion>
    <exclusion>
      <groupId>org.slf4j</groupId>
      <artifactId>jul-to-slf4j</artifactId>
    </exclusion>
    <exclusion>
      <groupId>org.slf4j</groupId>
      <artifactId>jcl-over-slf4j</artifactId>
    </exclusion>
    <exclusion>
      <groupId>org.slf4j</groupId>
      <artifactId>log4j-over-slf4j</artifactId>
    </exclusion>
  </exclusions>
</dependency>

  <!-- slf4j implementation to forward logs to log4j 2.x -->
  <dependency>
    <groupId>org.apache.logging.log4j</groupId>
    <artifactId>log4j-slf4j2-impl</artifactId> <!-- for slf4j 1.7.x use log4j-slf4j-impl instead -->
    <!-- version is omitted since there's org.apache.logging.log4j:log4j-bom in the dependencyManagement section -->
  </dependency>

```

```

    <scope>runtime</scope>
  </dependency>

  <!-- log4j 2.x bridges to forward java.util.logging, jcl/commons-logging and log4j 1.2.x to log4j 2.x -->
  <dependency>
    <groupId>org.apache.logging.log4j</groupId>
    <artifactId>log4j-jul</artifactId>
    <!-- version is omitted since there's org.apache.logging.log4j:log4j-bom in the dependencyManagement section -->
  </dependency>
  <dependency>
    <groupId>org.apache.logging.log4j</groupId>
    <artifactId>log4j-jcl</artifactId>
    <!-- version is omitted since there's org.apache.logging.log4j:log4j-bom in the dependencyManagement section -->
  </dependency>
  <dependency>
    <scope>runtime</scope>
  </dependency>
  <dependency>
    <groupId>org.apache.logging.log4j</groupId>
    <artifactId>log4j-1.2-api</artifactId>
    <!-- version is omitted since there's org.apache.logging.log4j:log4j-bom in the dependencyManagement section -->
  </dependency>
  <scope>runtime</scope>
</dependency>

  <!-- logging backend: log4j 2.x -->
  <!-- this dependency declarations are optional since org.apache.logging.log4j:log4j-slf4j-impl depends on them transitively -->
  <dependency>
    <groupId>org.apache.logging.log4j</groupId>
    <artifactId>log4j-api</artifactId>
    <!-- version is omitted since there's org.apache.logging.log4j:log4j-bom in the dependencyManagement section -->
  </dependency>
  <dependency>
    <groupId>org.apache.logging.log4j</groupId>
    <artifactId>log4j-core</artifactId>
    <!-- version is omitted since there's org.apache.logging.log4j:log4j-bom in the dependencyManagement section -->
  </dependency>
  <scope>runtime</scope>
</dependency>
</dependencies>

```

Example configuration for Apache Tika 2.5.0+ (Gradle)

Common sections

```

dependencies {
    // Import Maven BOM for Tika and Log4j 2.x.
    // Depending on your setup `api` could be used instead of `implementation` when `java-library` plugin is
    // activated.
    implementation(platform("org.apache.tika:tika-bom:2.6.0"))
    implementation(platform("org.apache.logging.log4j:log4j-bom:2.19.0"))

    constraints {
        // versions from constraints work like a dependencyManagement section in Maven
        implementation("org.slf4j:slf4j-api:2.0.3")
        implementation("org.slf4j:jul-to-slf4j:2.0.3")
        implementation("org.slf4j:jcl-over-slf4j:2.0.3")
        implementation("org.slf4j:log4j-over-slf4j:2.0.3")
    }

    implementation("org.apache.tika:tika-core")
    implementation("org.apache.tika:tika-parsers-standard-package")
}

configurations.all {
    // remove if using Apache Log4j 2.x log4j-jcl native bridge instead of jcl-over-slf4j
    exclude("commons-logging", "commons-logging")
}

```

Apache Log4j 2.x with slf4j bridges

```
// merge with common section above
dependencies {
    // versions from platform/BOM
    implementation("org.apache.logging.log4j:log4j-api")
    runtimeOnly("org.apache.logging.log4j:log4j-core")
    runtimeOnly("org.apache.logging.log4j:log4j-slf4j2-impl") // for slf4j 1.7.x use log4j-slf4j-impl instead

    implementation("org.slf4j:jul-to-slf4j") // java.util.logging to slf4j, requires additional configuration, see https://www.slf4j.org/api/org/slf4j/bridge/SLF4JBridgeHandler.html
    runtimeOnly("org.slf4j:jcl-over-slf4j") // commons-logging (JCL) to slf4j
    runtimeOnly("org.slf4j:log4j-over-slf4j") // log4j 1.2.x to slf4j
}

```

Logback

```
// merge with common section above
dependencies {
    constraints {
        // 1.4.x for slf4j 2.x & Jakarta EE 9+, 1.3.x for slf4j 2.x & Jakarta EE 8/Java EE 8, and 1.2.x for slf4j 1.7.
x        implementation("ch.qos.logback:logback-core:1.4.4")
        implementation("ch.qos.logback:logback-classic:1.4.4")
    }

    runtimeOnly("ch.qos.logback:logback-classic") // slf4j logging backend

    implementation("org.slf4j:jul-to-slf4j") // java.util.logging to slf4j, requires additional configuration, see https://www.slf4j.org/api/org/slf4j/bridge/SLF4JBridgeHandler.html
    runtimeOnly("org.slf4j:jcl-over-slf4j") // commons-logging (JCL) to slf4j
    runtimeOnly("org.slf4j:log4j-over-slf4j") // log4j 1.2.x to slf4j
    runtimeOnly("org.apache.logging.log4j:log4j-to-slf4j") // log4j 2.x to slf4j adapter
}

```

Apache Log4j 2.x with native bridges

```
dependencies {
    // versions from platform/BOM
    implementation("org.apache.logging.log4j:log4j-api")
    runtimeOnly("org.apache.logging.log4j:log4j-core")
    runtimeOnly("org.apache.logging.log4j:log4j-slf4j2-impl") // for slf4j 1.7.x use log4j-slf4j-impl instead

    runtimeOnly("org.apache.logging.log4j:log4j-jul") // java.util.logging to log4j 2.x adapter, requires
additional configuration see https://logging.apache.org/log4j/2.x/log4j-jul/index.html
    runtimeOnly("org.apache.logging.log4j:log4j-jcl") // commons-logging (JCL) to log4j 2.x bridge
    runtimeOnly("org.apache.logging.log4j:log4j-1.2-api") // log4j 1.2.x to log4j 2.x bridge
}

```