(Due to technical issues, the search service is temporarily unavailable.)

To efficiently build your Spring Batch project with Jasper Reports 6.23 and automate the compilation of `.jrxml` files to `.jasper`, follow these best practices:

### 1. \*\*Project Structure\*\*

Organize your `.jrxml` files in a dedicated directory (e.g., `src/main/resources/jasperreports`). Compiled `.jasper` files should be placed in `target/classes/jasperreports` to be included in the classpath.

```

src/

main/

resources/

jasperreports/ <-- .jrxml files here

```

### 2. \*\*Maven Configuration\*\*

#### a. \*\*Jasper Reports Dependency\*\*

Add the Jasper Reports dependency to your `pom.xml`:

```xml

<dependencies>

<dependency>

<groupId>net.sf.jasperreports</groupId>

<artifactId>jasperreports</artifactId>

<version>6.23.0</version>

</dependency>

<!-- Additional dependencies (e.g., fonts, exports) -->

<dependency>

<groupId>net.sf.jasperreports</groupId>

<artifactId>jasperreports-fonts</artifactId>

<version>6.23.0</version>

</dependency>

</dependencies>

```

#### b. \*\*JasperReports Maven Plugin\*\*

Use the `jasperreports-maven-plugin` to compile `.jrxml` files during the build:

```xml

<build>

<plugins>

<plugin>

<groupId>org.codehaus.mojo</groupId>

<artifactId>jasperreports-maven-plugin</artifactId>

<version>2.8</version> <!-- Check for latest compatible version -->

<executions>

<execution>

<phase>generate-sources</phase>

<goals>

<goal>compile-reports</goal>

</goals>

</execution>

</executions>

<configuration>

<sourceDirectory>src/main/resources/jasperreports</sourceDirectory>

<outputDirectory>${project.build.outputDirectory}/jasperreports</outputDirectory>

<compiler>net.sf.jasperreports.engine.design.JRJdtCompiler</compiler>

<encoding>UTF-8</encoding> <!-- For consistent encoding -->

</configuration>

<dependencies>

<!-- Match Jasper Reports version with your project -->

<dependency>

<groupId>net.sf.jasperreports</groupId>

<artifactId>jasperreports</artifactId>

<version>6.23.0</version>

</dependency>

</dependencies>

</plugin>

<!-- Configure JDK 21 compatibility -->

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.11.0</version>

<configuration>

<source>21</source>

<target>21</target>

<encoding>UTF-8</encoding>

</configuration>

</plugin>

</plugins>

</build>

```

### 3. \*\*Build Process\*\*

- Run `mvn clean install` to:

1. Compile `.jrxml` files into `.jasper` during the `generate-sources` phase.

2. Package the application with compiled reports in the classpath.

### 4. \*\*Loading Reports in Spring Batch\*\*

Use the classpath to load compiled `.jasper` files:

```java

Resource resource = new ClassPathResource("jasperreports/your\_report.jasper");

JasperReport report = (JasperReport) JRLoader.loadObject(resource.getInputStream());

```

### 5. \*\*Best Practices\*\*

- \*\*Version Control\*\*: Exclude `.jasper` files from Git (add `\*\*/\*.jasper` to `.gitignore`).

- \*\*Error Handling\*\*: The plugin will fail the build if there are errors in `.jrxml` files.

- \*\*IDE Integration\*\*: Refresh your IDE after compilation to see generated files.

### 6. \*\*Verification\*\*

Check the `target/classes/jasperreports` directory after a build to confirm `.jasper` files are generated.

---

This setup ensures your Jasper Reports are compiled automatically during the build process and integrates seamlessly with Spring Batch and JDK 21.

To efficiently build your Spring Batch project with Jasper Reports 6.23 and automate the compilation of .jrxml files to .jasper, follow these best practices:

### **1. Project Structure**

Organize your .jrxml files in a dedicated directory (e.g., src/main/resources/jasperreports). Compiled .jasper files should be placed in target/classes/jasperreports to be included in the classpath.

Copy

src/

main/

resources/

jasperreports/ <-- .jrxml files here

### **2. Maven Configuration**

#### **a. Jasper Reports Dependency**

Add the Jasper Reports dependency to your pom.xml:

xml

Copy

<dependencies>

<dependency>

<groupId>net.sf.jasperreports</groupId>

<artifactId>jasperreports</artifactId>

<version>6.23.0</version>

</dependency>

<!-- Additional dependencies (e.g., fonts, exports) -->

<dependency>

<groupId>net.sf.jasperreports</groupId>

<artifactId>jasperreports-fonts</artifactId>

<version>6.23.0</version>

</dependency>

</dependencies>

Run HTML

#### **b. JasperReports Maven Plugin**

Use the jasperreports-maven-plugin to compile .jrxml files during the build:

xml

Copy

<build>

<plugins>

<plugin>

<groupId>org.codehaus.mojo</groupId>

<artifactId>jasperreports-maven-plugin</artifactId>

<version>2.8</version> <!-- Check for latest compatible version -->

<executions>

<execution>

<phase>generate-sources</phase>

<goals>

<goal>compile-reports</goal>

</goals>

</execution>

</executions>

<configuration>

<sourceDirectory>src/main/resources/jasperreports</sourceDirectory>

<outputDirectory>${project.build.outputDirectory}/jasperreports</outputDirectory>

<compiler>net.sf.jasperreports.engine.design.JRJdtCompiler</compiler>

<encoding>UTF-8</encoding> <!-- For consistent encoding -->

</configuration>

<dependencies>

<!-- Match Jasper Reports version with your project -->

<dependency>

<groupId>net.sf.jasperreports</groupId>

<artifactId>jasperreports</artifactId>

<version>6.23.0</version>

</dependency>

</dependencies>

</plugin>

<!-- Configure JDK 21 compatibility -->

<plugin>

<groupId>org.apache.maven.plugins</groupId>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.11.0</version>

<configuration>

<source>21</source>

<target>21</target>

<encoding>UTF-8</encoding>

</configuration>

</plugin>

</plugins>

</build>

Run HTML

### **3. Build Process**

* Run mvn clean install to:
  1. Compile .jrxml files into .jasper during the generate-sources phase.
  2. Package the application with compiled reports in the classpath.

### **4. Loading Reports in Spring Batch**

Use the classpath to load compiled .jasper files:

java

Copy

Resource resource = new ClassPathResource("jasperreports/your\_report.jasper");

JasperReport report = (JasperReport) JRLoader.loadObject(resource.getInputStream());

### **5. Best Practices**

* **Version Control**: Exclude .jasper files from Git (add \*\*/\*.jasper to .gitignore).
* **Error Handling**: The plugin will fail the build if there are errors in .jrxml files.
* **IDE Integration**: Refresh your IDE after compilation to see generated files.

### **6. Verification**

Check the target/classes/jasperreports directory after a build to confirm .jasper files are generated.

This setup ensures your Jasper Reports are compiled automatically during the build process and integrates seamlessly with Spring Batch and JDK 21.