

Reporting Encoding Configuration

Currently, the character encoding for reports output files needs to be configured individually for each and every plugin that creates new report types (i.e. don't write its content through Doxia but write directly its files to disk).

Life would become easier if there was a dedicated POM element like `${project.reporting.outputEncoding}` which could be used to specify the encoding once per entire project. Every plugin could use it as default value, like it has been done with [source files encoding](#):

```
/**
 * @parameter expression="${outputEncoding}"
 * default-value="${project.reporting.outputEncoding}"
 */
private String outputEncoding;
```

Adding this element to the POM structure without breaking backward compatibility can only happen in a future version, yet to be defined (at least after Maven 3.0):

```
<project>
  ...
  <reporting>
    <outputEncoding>UTF-8</outputEncoding>
    ...
  </reporting>
  ...
</project>
```

For Maven 2.x, the value can be defined as an equivalent property:

```
<project>
  ...
  <properties>

<project.reporting.outputEncoding>UTF-8</pro
ject.reporting.outputEncoding>
  ...
</properties>
  ...
</project>
```

Thus plugins could immediately be modified to use `${project.reporting.outputEncoding}` expression, whatever Maven version is used.

Default Value

Actually, default output encoding vary between plugins:

- ISO-8859-1 for maven-site-plugin, maven-jxr-plugin and by extension every reporting plugin generating content with maven-site-plugin's template (that is the vast majority of reporting plugins),
- UTF-8 for cobertura-maven-plugin,
- platform encoding for maven-javadoc-plugin.

Unifying default value will lead to a change for plugins previously using another default value. This shouldn't cause much harm since reports are mainly read by humans through their web browser.

After a [poll on user list](#), the default value is **UTF-8**, which will ensure that default value is appropriate for characters in any language in the world.

A check has to be coded in every plugin with the default value:

```

/**
 * Gets the effective reporting output files
 encoding.
 *
 * @return The effective reporting output
 file encoding, never <code>null</code>.
 */
protected String getOutputEncoding()
{
    return ( outputEncoding != null ) ?
outputEncoding : ReaderFactory.UTF_8;
}

```

This default value can be coded in POM model too when available (default value of the outputEncoding attribute) and in super-pom in Maven 2.x. But this change is only for clarity since without it, the previous check coded in every plugin will transform *null* value to the chosen default value.

For the record, the other proposed unified default value was **source encoding** as defined in [Source File Encoding](#) proposal, which would vary from project to project. Since users are invited to set a fixed value for source encoding in their poms to ensure build reproducibility, such calculated value wouldn't affect build reproducibility, but wouldn't have been appropriate in every case (for example, if project's description in pom.xml contains characters that can't be output with source files encoding).

Plugins to Modify

The vast majority of reporting plugins don't need any change since they are using Doxia and maven-site-plugin's template: the encoding configuration will silently be inherited from maven-site-plugin.

Affected Apache plugins:

- maven-changelog-plugin
- maven-javadoc-plugin: [MJAVADOC-206](#), done in 2.5
- maven-jxr-plugin: [JXR-67](#), done in 2.2-SNAPSHOT
- maven-pmd-plugin: [MPMD-83](#)
- maven-site-plugin: [MSITE-340](#), done in 2.0

Affected Codehaus plugins:

- cobertura-maven-plugin

References

Please see [0] for the related thread from the mailing list and [1] for the corresponding feature request in JIRA.

[0] [next step for encoding support: reporting output files](#)

[1] [MNG-3608](#)