Versions Maven Plugin

The Versions Plugin is used when you want to manage the versions of artifacts in a project's POM.

Goals Overview

The Versions Plugin has the following goals.

- versions:compare-dependencies compares the dependency versions of the current project to the dependency management section of a remote project.
- versions:display-dependency-updates scans a project's dependencies and produces a report of those dependencies which have newer versions available.
- versions:display-plugin-updates scans a project's plugins and produces a report of those plugins which have newer versions available.
- versions:display-property-updates scans a projectand produces a report of those properties which are used to control artifact versions and which properies have newer versions available.
- versions:update-parent updates the parent section of a project so that it references the newest available version. For example, if you use a corporate root POM, this goal can be helpful if you need to ensure you are using the latest version of the corporate root POM.
- versions:update-properties updates properties defined in a project so that they correspond to the latest available version of specific dependencies. This can be useful if a suite of dependencies must all be locked to one version.
- versions:update-property Sets a property to the latest version in a given range of associated artifacts.
- versions:update-child-modules updates the parent section of the child modules of a project so the version matches the version of the current project. For example, if you have an aggregator pom that is also the parent for the projects that it aggregates and the children and parent versions get out of sync, this mojo can help fix the versions of the child modules. (Note you may need to invoke Maven with the -N option in order to run this goal if your project is broken so badly that it cannot build because of the version mis-match).
- versions:lock-snapshots searches the pom for all -SNAPSHOT versions and replaces them with the current timestamp version of that -SNAPSHOT, e.g. -20090327.172306-4
- versions:unlock-snapshots searches the pom for all timestamp locked snapshot versions and replaces them with -SNAPSHOT.
- versions:resolve-ranges finds dependencies using version ranges and resolves the range to the specific version being used.
- versions:set can be used to set the project version from the command line.
- versions:use-releases searches the pom for all -SNAPSHOT versions which have been released and replaces them with the corresponding release version.
- versions:use-next-releases searches the pom for all non-SNAPSHOT versions which have been a newer release and replaces them with the next release version.
- versions:use-latest-releases searches the pom for all non-SNAPSHOT versions which have been a newer release and replaces them with the latest release version.
- versions:use-next-snapshots searches the pom for all non-SNAPSHOT versions which have been a newer -SNAPSHOT version and replaces them with the next -SNAPSHOT version.
- versions:use-latest-snapshots searches the pom for all non-SNAPSHOT versions which have been a newer -SNAPSHOT version and replaces them with the latest -SNAPSHOT version.
- versions:use-next-versions searches the pom for all versions which have been a newer version and replaces them with the next version.
- versions:use-latest-versions searches the pom for all versions which have been a newer version and replaces them with the latest version.
- versions:commit removes the `pom.xml.versionsBackup` files. Forms one half of the built-in "Poor Man's SCM".

- versions:revert restores the `pom.xml` files from the `pom.xml.versionsBackup` files. Forms one half of the built-in "Poor Man's SCM".

Reporting goals overview

The Versions Plugin has the following reporting goals.

- versions:dependency-updates-report produces a report of those project dependencies which have newer versions available.
- versions:plugin-updates-report produces a report of those plugins which have newer versions available.
- versions:property-updates-report produces a report of those properties which are used to control artifact versions and which properies have newer versions available.

Usage

General instructions on how to use the Versions Plugin can be found on the usage page. Some more specific use cases are described in the examples given below. Last but not least, users occasionally contribute additional examples, tips or errata to the plugin's wiki page.

In case you still have questions regarding the plugin's usage, please have a look at the FAQ and feel free to contact the user mailing list. The posts to the mailing list are archived and could already contain the answer to your question as part of an older thread. Hence, it is also worth browsing/searching the mail archive.

If you feel like the plugin is missing a feature or has a defect, you can fill a feature request or bug report in our issue tracker. When creating a new issue, please provide a comprehensive description of your concern. Especially for fixing bugs it is crucial that the developers can reproduce your problem. For this reason, entire debug logs, POMs or most preferably little demo projects attached to the issue are very much appreciated. Of course, patches are welcome, too. Contributors can check out the project from our source repository and will find supplementary information in the guide to helping with Maven.

Examples

To provide you with better understanding of some usages of the Plugin Name, you can take a look into the following examples:

- Advancing dependency versions
- Compare project dependencies to a remote project
- Checking for new dependency updates
- Checking for new plugin updates
- Checking for new property-linked updates
- Updating the Parent version
- Updating a version specified in a property
- Fixing a multi-module build
- Resolve version ranges
- Locking snapshot dependencies
- Unlocking snapshot dependencies
- Replacing -SNAPSHOT versions with their corresponding releases
- Changing the project version