

Version Cleaner

The version cleaner module provides commands to remove old or duplicated versions from your version store. Which versions are selected for removal can be configured through voters.

Analyzing and removing unneeded versions can take a lot of time, the version cleanup command can be configured to examine a set number of versions (actually nodes in the version workspace) and then stop. The next run of the version cleanup command will resume from the node last analyzed.

The version cleaner can be configured to select which versions are deleted or retained.

JIRA	-
Git	versioncleaner

Installation

Maven is the easiest way to install the module. Add the following dependency to your [bundle](#):

```
<dependency>
  <groupId>info.magnolia.services</groupId>
  <artifactId>versioncleaner</artifactId>
  <version>${versioncleanerVersion}</version>
</dependency>
```

Versions

1.6.2	Magnolia 5.7+
1.6.1	Magnolia 5.7+
1.6	Magnolia 5.7+
1.5	Magnolia 5.7+

Usage

The module provides a set of commands for removing selected versions from the version history.

Which versions are selected for deletion are configurable through voters. You can define complex conditions to identify if a version should be deleted.

Cleanup Commands

The cleanup commands can be found in the version cleaner [catalog](#): `/module/versioncleaner/commands/cleanup`

<code>cleanVersions</code>	Remove unneeded versions according to its configuration.
<code>cleanAllVersions</code>	Remove all versions found except root and leaf versions. Unlike <code>cleanVersions</code> , <code>cleanAllVersions</code> does not require possibly complicated configuration.
<code>cleanNodeVersions</code>	Clean versions from a workspace and path.

`cleanVersions` and `cleanAllVersions` work directly on the JCR version workspace. They clean versions from all workspaces with versions.

`cleanNodeVersions` however cleans versions from a specific workspace and optional path. Unlike `cleanVersions` and `cleanAllVersions`, `cleanNodeVersions` will not resume its cleaning run after reaching its `visitLimit`; it doesn't save its progress while cleaning versions. `cleanNodeVersions` does not archive versions before deleting them.

The `cleanVersions` and `cleanAllVersions` commands can be configured with the following parameters when invoked:

noop	true or false (the default) Does a "dry run", going through the selected versions but without deleting or modifying them.
archive	true or false (the default) If true, will save an export of each version to be deleted to the specified <code>archiveDir</code> . NOTE: using this option may generate many files. Be sure you have sufficient space to store the archive versions.
archiveDir	Directory for storing archived versions and cleaning run history. Defaults to: <code>\${magnolia.cleanup.dir}</code> if specified or <code>\${magnolia.repositories.home}</code> if not. NOTE: Magnolia must have read and write permissions for the directory.
visitLimit	The maximum number of version nodes that will be checked during a run. Defaults to 1000 if not specified. If set to a number less than 0, all version nodes will be checked.

These parameters can also be set as properties in the command configuration, at `/module/versioncleaner/commands/cleanup/cleanVersions` or `/module/versioncleaner/commands/cleanup/cleanAllVersions`.

Diagnostic Commands

The diagnostic commands can be found in the version cleaner [catalog](#): `/module/versioncleaner/commands/diagnostic`

findBadVersions	Locate corrupted versions in the version workspace. A version is corrupt if it: <ul style="list-style-type: none"> ■ has no root version ■ has multiple root versions ■ the root version identified by name differs from the root version returned by the JCR API
-----------------	---

Configuring Cleanup

The versions selected for removal by the `cleanVersions`, `cleanAllVersions` and `cleanNodeVersions` commands are controlled by [Voters](#). The voters determine if a versions should be deleted (the result of the voting is true) or retained (the result of the voting is false).

You can combine the results of different voters with logical operations (and, or, xor...).

Each command can be configured with the `pruneVersions` property:

Node name	Value
modules	
versioncleaner	
commands	
cleanup	
cleanVersions	
pruneVersion	
oldVersion	
noModification	
class	<code>info.magnolia.voting.voters.VoterSet</code>
op	or

Version Voters

You can use the following voters to identify versions to be deleted or retained:

EligibleVersionVoter

EligibleVersionVoter returns true if the version is not a root version (the first version of a node) or a leaf version (the last version of a node).

Class: `info.magnolia.services.cleanup.commands.voters.EligibleVersionVoter`

NoModificationVoter

NoModificationVoter returns true if the version has the same modification date as the previous version or false if the modification dates are different.

Class: `info.magnolia.services.cleanup.commands.voters.NoModificationVoter`

OldVersionVoter

OldVersionVoter returns true if the version is older than the `maxAge` threshold or false otherwise.

Class: `info.magnolia.services.cleanup.commands.voters.OldVersionVoter`

Properties:

<code>maxAge</code>	optional The age threshold in milliseconds Defaults to 90 days (i.e. 7776000000)
---------------------	--

UnneededVersionVoter

UnneededVersionVoter returns true if there are at least `versionsThreshold` newer versions and false otherwise. Use `UnneededVersionVoter` to limit the total number of versions retained.

Class: `info.magnolia.services.cleanup.commands.voters.UnneededVersionVoter`

Properties:

<code>versionsThreshold</code>	optional The number of versions to be retained Defaults to 5
--------------------------------	--

Warnings

- This module is at INCUBATOR level.

Changelog

- v1.6.2
 - Corrected ordering of version workspace traversal
- v1.6.1
 - Corrected naming for pruning configuration for `info.magnolia.services.cleanup.commands.NodeVersionCleanupCommand` (`pruneVersions` → `pruneVersion`)
- v1.6
 - Removed obsolete `pruneHistory` configuration
 - Corrected resumption of pruning from history
 - Added `cleanNodeVersions` command (`info.magnolia.services.cleanup.commands.NodeVersionCleanupCommand`)