Build Files Cleaner (Swabra)

Swabra (originally from the Russian noun 'shvabra' - a mop, also from the English verb 'swab' - clean with a mop) is a bundled plugin allowing you to clean files created during the build.

The plugin remembers the state of the file tree after the sources checkout and deletes all the newly added files at the end of the build or at the next build start depending on the settings. Swabra also detects files modified or deleted during the build and reports them to the build log (however, such files are not restored by the plugin). The plugin can also ensure that by the start of the build there are no files modified or deleted by previous builds and initiate clean checkout if such files are detected.

Moreover, Swabra gives the ability to dump processes which lock directory by the end of the build (requires handle.exe)

Swabra can be added as a build feature to your build configuration regardless of what set of build steps you have. By configuring its options you can enable scanning the checkout directory for newly created, modified and deleted files and enable file locking processes detection.

> ⓘ Swabra should be used with the automatic checkout only: after this build feature is configured, it will run **before the first build step** to remember the state of the file tree after the sources checkout and to restore it after the build.

The checkout directory state is saved into a file in the caches directory named `<checkout_directory_name_hash>.snapshot` using the DiskDir format. The path to the checkout directory to be cleaned is saved into the `snapshot.map` file. The snapshot is used later (at the end of the build or at the next build start) to determine which files and folders are newly created, modified or deleted. It is done based on the actual files' presence, last modification data and size comparison with the corresponding records in the snapshot.

Configuring Swabra Options

Option	Description
Files cleanup	Select whether you want to perform build files cleanup, and when it will be performed.
Clean checkout	Select the Force clean checkout if cannot restore clean directory state option to ensure that the checkout directory corresponds to the sources in the repository at the build start. If Swabra detects any modified or deleted files in the checkout directory before the build start, it will enforce clean checkout. The build will fail if Swabra cannot delete some files created during the previous build. If this option is disabled, you will only get warnings about modified and deleted files.
Paths to monitor	Specify a newline-separated set of `+|-:path` rules to define which files and folders are to be involved in the files collection process (by default and until explicitly excluded, the entire checkout directory is monitored). The path can be relative (based on the build's checkout directory) or absolute and can include Ant-like wildcards. If no `+:` or `-:` prefix is specified, a rule as treated as "include". Rules on any path must come in the order from more general to more concrete. The top level path must always point to a directory. Specifying a directory affects its entire content and sub-directories. Note also that Swabra is case-sensitive. Examples: `-:*/dir/*` excludes all `dir` folders and their content, `-:some/dir, +:some/dir/inner` excludes `some/dir` folder and all its content except for the `inner` subfolder and its content `+:./*file.txt` includes only the specified file in the build checkout directory into monitoring `-:file.txt` excludes the specified file in the build checkout directory from monitoring ⚠ Note that after removing some exclude rules, it is advisable to run a clean checkout.
Locking processes	Select whether you want Swabra to inspect the checkout directory for processes locking files in this directory, and what to do with such processes. Note that handle.exe is required on agents for locking processes detection.

Verbose output	Check this option to enable detailed logging to build log.

Default excluded paths

If the build is set up to checkout on the agent, by default Swabra ignores all `.svn`, `.git`, `.hg`, `CVS` folders and their content.
To turn off this behaviour, specify an empty `swabra.default.rules` configuration parameter.

Installing Handle

You can install `handle.exe` from the Administration | Tools page.
Click the Install Tool button and select Sysinternals handle.exe from the list of tools.
Specify whether you want to download the latest version of handle.exe or upload it manually choosing the path on local machine, and click Add. TeamCity will download or upload `handle.exe` and send it to Windows agents.

handle.exe is present on agents only after the upgrade.

Please note that running handle.exe requires some additional permissions for the build agent user. For more details please read this thread.

Debug options

Generally snapshot file is deleted after files collection. Set the `swabra.preserve.snapshot` system property to preserve snapshots for debugging purposes.

Clean Checkout

Please note that Swabra may sometimes cause clean checkout to restore clean checkout directory state.
To avoid unnecessary frequent clean checkouts, always set up identical Swabra build features for build configurations working in the same checkout directory.
Build configurations work in the same checkout directory if either the same custom checkout directory path or same VCS settings configured for them.

Development links

See plugin page at Swabra.