DICE PLUGIN USER MANUAL

1. Setting up Post-Build Actions

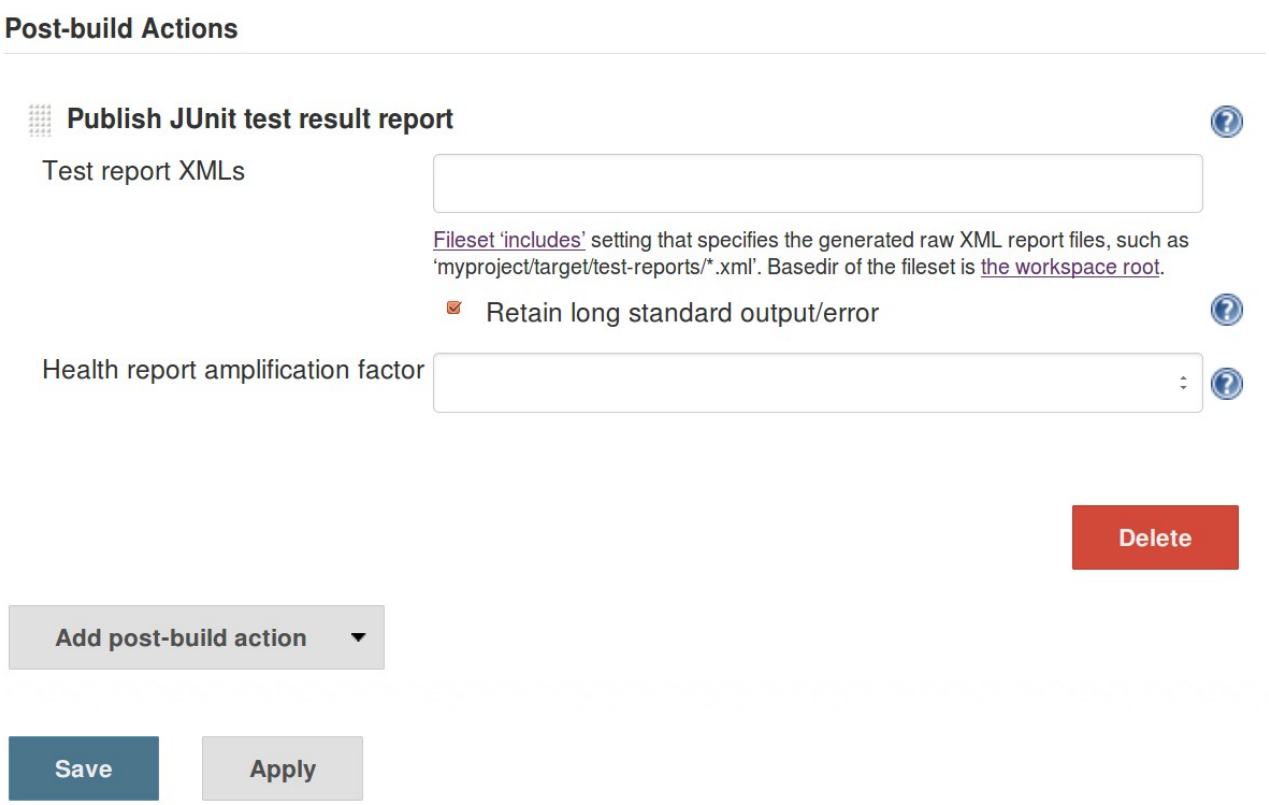
In order for the DICE plugin to display the results of your testing tools, you need to provide the path to the directory that contains these results. Since that directory can contain files other than the results that you wish the DICE plugin to display, you can also provide a name pattern in order to only read and display certain files (i.e. your result files).

From the Jenkins **Dashboard**, click the **job** you wish to configure.

From there, click **Configure** in the menu on the left side of the screen.

If you haven't added a Post-Build Action yet, click the **Add post-build action** dropdown menu, and click **Publish DICE test result report**.

You will see a newly created “Publish DICE test result report” section (see picture below).



In the **Test report XMLs**, specify the path to your result files the Ant glob syntax, such as `**/build/test-reports/*.xml`. Be sure not to include any non-report files into this pattern. You can specify multiple patterns of files separated by commas.

If you check the **Retain long standard out/error** checkbox, any standard output or error from a test suite will be retained in the test results after the build completes. (This refers only to additional messages printed to console, not to a failure stack trace.) Such output is always kept if the test

failed, but by default lengthy output from passing tests is truncated to save space. Check this option if you need to see every log message from even passing tests, but beware that Jenkins's memory consumption can substantially increase as a result, even if you never look at the test results!

In the **Health report amplification factor**, provide the amplification factor to apply to test failures when computing the test result contribution to the build health score. The default factor is `1.0`.

- A factor of `0.0` will disable the test result contribution to build health score.
- A factor of `0.1` means that 10% of tests failing will score 99% health
- A factor of `0.5` means that 10% of tests failing will score 95% health
- A factor of `1.0` means that 10% of tests failing will score 90% health
- A factor of `2.0` means that 10% of tests failing will score 80% health
- A factor of `2.5` means that 10% of tests failing will score 75% health
- A factor of `5.0` means that 10% of tests failing will score 50% health
- A factor of `10.0` means that 10% of tests failing will score 0% health

The factor is persisted with the build results, so changes will only be reflected in new builds.

When you're done, click **Save**.