

# eraseBuild

The *eraseBuild* program is invoked by the [releaseManagerDaemon](#) program to completely remove older builds from disk. It can also be invoked manually (usually via the [deleteBuild](#) program) to remove a bad build.

When invoked, this program requires the `--buildId` flag to be specified with the **buildId** corresponding to a valid entry in the [build](#) database table. Thus all invocations must be of the form:

```
eraseBuild --buildId <buildID>
```

where `<buildId>` is the build package to be worked on. In addition to the `--buildId` flag, the *eraseBuild* program also accepts the `--complete` flag (with no arguments). By default, the *eraseBuild* program removes the files associated with the build but leaves the database entries giving the information on compile and test status. If the `--complete` flag is specified, the database entry is removed as well which will usually result in retriggering the build via the [releaseManagerDaemon](#) program.

When invoked the *eraseBuild* program does the following:

1. Removes the directory specified by the **buildLocation** field in the [build](#) table for this **buildId**
2. Clears the **buildLocation**, **userReleaseLocation**, and **sourceReleaseLocation** fields in the [build](#) table
3. For any build that has **distributionLocation** set in the [settings](#) table, the user, source, and devel distribution packages are explicitly deleted. On the Linux systems these packages are in the **buildLocation** directory and will be removed when that directory is removed. However, for builds that have **distributionLocation** set, they are not and must be explicitly removed.

If the `--complete` option is specified, the entry for the build in the [build](#) table is copied to the [deletedBuild](#) table and then removed from the [build](#) table. This allows the information on the earlier build to be preserved but clears the way for a new build to be run. The `--complete` option is typically specified when a build failed due to infrastructure reasons so that the Release Manager will retrigger the build so that it can build correctly.

## Notes:

- *This program does not currently run for any builds built by the Jenkins system as there has been no mechanism set up to trigger this program through Jenkins. It will, however, run properly if invoked manually on the Windows and Linux systems. It will mostly work on the Mac but will not remove the distribution packages as they are not directly accessible via the file system. Rather functionality has to be added to remove them via an ssh command.*
- *The program clears **buildLocation**, **userReleaseLocation** and **sourceReleaseLocation** but not **develReleaseLocation** in the build table. This should probably be corrected.*