

Release process for Subplot

The Subplot project

2021-10-21 11:32

Contents

The Subplot project consists of several crates. They can be released independently from each other. However, there is only one `NEWS.md` file with release notes, and only one set of Debian packaging (only the binaries and template resources are packaged). The crates are, currently:

- `subplot-build`
- `subplotlib-derive`
- `subplotlib`
- `subplot`

To find all crates, run the following at the root of the source tree:

```
git ls-files | grep Cargo.toml
```

Do these for each crate, with leaf crates in the dependency tree first:

1. Check that the dependency versions on any other Subplot crates are correct.
2. Run `cargo update` and `cargo outdated` for the crate, and make any needed updates.
3. Review changes in the crate (`git log -p .` in the crate's sub-directory). Update the top level `NEWS.md` with any changes that users of Subplot need to be aware of.
4. Update the crate's `Cargo.toml` with the appropriate version number for the new release, if there's been any changes. If any of the other crates depend on this crate, update their dependency information in their `Cargo.toml` as needed.
5. Run `cargo publish --dry-run` and fix any problems.

For the top crate `subplot` additionally do the following:

1. Run `./check` and fix anything that needs fixing to make it finish successfully.
2. Update `NEWS.md` with an end-user oriented summary of the changes in all crates since the previous release, if there are any.

3. Update `debian/changelog` with a summary of any changes to the Debian packaging. Use the `dch` command to edit the file to get the format right: `dch -v X.Y.Z-1 "New release"` to start a new version, `dch "foo bar"` to add a new bullet point to a version, and `dch -r ""` to mark the version as ready for release.
4. Commit the changes in all crates and submit as a merge request via GitLab.

Once the above changes have been merged, make the actual release (note that these steps are meant to be possible to automate, later on, and require no review cycle):

1. Pull down the above changes from GitLab.
2. Create a signed, annotated, git tag `X.Y.Z` for version `X.Y.Z` of the Subplot project (`git tag -sam "Subplot release X.Y.Z" X.Y.Z`). We don't tag releases of the other crates.
3. Run `git push --tags gitlab` to push the tag to Gitlab. Replace `gitlab` with whatever name you use in git for the remote that is `gitlab.com`.
4. Run `cargo publish` in any crates that have changed since the previous release, from the leaves toward the root.
5. Ask Lars or Daniel to pull the tag from GitLab and push it to `git.liw.fi` so that a Debian package gets built and published.
6. Announce our jubilation to the world via blog posts and other suitable channels.