L07 Maintenance

Markus Raab

Institute of Information Systems Engineering, TU Wien

Maintenance

Learning Outcomes

After successful completion of L07
students will be able to use techniques to

- maintain FLOSS initiatives

Releases

- main communication to the "outside world"
- tag which version to use (e.g. when some feature is ready)
- points in time when to test, package and announce

Release Notes

- communicate what changed
- say thanks to contributors
- give incentive to try out the latest version
- give maintainers hints how to package
- communicate API changes to developers

Release Schema

If possible, avoid them, e.g. use date as release number.
If you want to convey some semantic meaning, like:

- compatibility in APIs (major.minor.micro)
- stable/unstable branches (even/odd, alpha/beta) or LTS

then a different release schema might make sense.

Licensing

- choose one license for your whole initiative
- if you copy code from others:
    - it either has the same license
    - it has a compatible license AND you document all licenses you use (reuse.software)

Source Releases

- tar the source code
- sign the tar ball

$\rightarrow$ gets less important
distributions now also pull from version control

Binary Releases

important if you need to directly publish to end users:

- compile on some old distributions and tar the binaries
- flatpack, homebrew, 0install

Linux Distributions

- packages maintained by the distributions
- can be considerable effort, so it is good if you can help by reporting:
  - which licenses are used (reuse)
  - which dependencies are needed
  - which files are installed

Continuous Releases

Automate:

- which installed files were added/removed
- publishing release notes on web-site
- creating and publishing packages
- tests
- generate version-specific information (like tested compilers)

L07 Maintenance

Markus Raab

Institute of Information Systems Engineering, TU Wien

Meeting

Learning Outcomes

After successful completion of L07
students will be able to use techniques to

- maintain FLOSS initiatives

Releases

Task

What purpose do releases have?

Releases

- main communication to the "outside world"
- tag which version to use (e.g. when some feature is ready)
- points in time when to test, package and announce

Release Notes

Task

What purpose do release notes have?

Release Notes

- communicate what changed
- say thanks to contributors
- give incentive to try out the latest version
- give maintainers hints how to package
- communicate API changes to developers

Release Notes

Task

Brainstorm how good release notes should be written.

Task

Break.

Licensing

Task

What needs to be considered in regards to licensing?

Licensing

- choose one license for your whole initiative
- if you copy code from others:
    - it either has the same license
    - it has a compatible license AND you document all licenses you use (reuse.software)

Licensing

Task

Which license do you (personally) prefer?

Task

Break.

Linux Distributions

Task

Who maintains packages in Linux distributions?
How can you help as FLOSS maintainer?

Linux Distributions

- packages maintained by the distributions
- can be considerable effort, so it is good if you can help by reporting:
    - which licenses are used (reuse)
    - which dependencies are needed
    - which files are installed

Linux Distributions

Task

Which FLOSS operating system did you try?
What was your experience?

ready to merge

set label "ready to merge" for all your PRs iff

- build server is happy
- you ticked all relevant points in PR

ping me (@markus2330) on any questions

H3+T2 reviews

- actually do testing (e.g. run shell recorder of PR)
- write what exactly you did (e.g. post output of your run)

P2

- documentation
- entry barriers
- literature research

L08 Collaboration

before holidays
with Rhonda