Contribution ID: **68**   Type: **Talk (15min + 5min)**

Three Lessons Learned: How RSEs Succeed in License Management

Tuesday 25 February 2025 15:10 (20 minutes)

Software license management is a critical but often overlooked aspect of Research Software Engineering (RSE). For both open-source and proprietary software projects, proper license management is increasingly important for sustainability, compliance, and collaboration. Our talk presents three key lessons learned from our experiences in license management, based on interdisciplinary projects and case studies at KIT. These lessons should help RSEs to overcome the challenges of license compliance in academic and industrial environments and to ensure long-term software value.

1. Generate Software Bill of Materials (SBOM) for Transparency

A key takeaway is the importance of creating and maintaining a Software Bill of Materials (SBOM) early on from the start in any RSE project. An SBOM provides a comprehensive inventory of all components and their associated licenses. It ensures transparency by clarifying which licenses apply to which parts of the code, and is especially valuable when collaborating with industry. In one case, a partner required software that had to be compliant with industry standards (e.g. ISO5230). The team had to do a lot of retrospective work to meet these requirements, highlighting the need for an SBOM from the beginning to avoid legal and financial complications later.

2. Carefully Manage Contributions

Using version management in your DevOps platform (e.g. GitLab) is essential for Research Software projects, among others to track the development process, coordinate between participating developers, and provide access to current and previous versions of the project. Within such a structured approach, it's also important to take care of license management, especially for handling incoming contributions.
Effective license management must extend beyond outbound licensing to the contributions RSEs accept from others. Inbound contributions must align with the project's outbound licensing strategy. For example, third-party contributions may introduce incompatible licenses, which can disrupt a project's legal position. This lesson emphasizes the need to carefully evaluate all external code to avoid issues like improperly licensed "snippets" from public forums. Tools like Fossology and REUSE help streamline this process by checking for license compliance, ensuring that all contributions are consistent with the project's overall license model.

3. Maintain your Flexibility to Adapt Your License Model to the Community

The third lesson is to remain adaptable in your licensing decisions. Different communities and industries may require different licensing strategies. In one project, an RSE team had to deal with dual licensing issues when an industry partner requested a non-copyleft version of RSE's "GPL-ed"software. By adapting their licensing model, they were able to serve both the open-source community and the proprietary software market. Such flexibility can extend the reach and value of the software, allowing RSEs to balance community engagement with commercialization opportunities.

Conclusion

By integrating these three lessons—generating SBOMs, carefully managing contributions, and maintaining flexible with licensing RSEs can navigate the complexities of license compliance. These strategies not only improve the sustainability of research software, but also open doors for broader collaboration and industry adoption. Our presentation will provide real-world examples, tools, and techniques to help RSEs master license management for long-term project success.

I want to participate in the youngRSE prize

Primary authors: DREES T.; FEUCHTER D.; STARY T.; WINANDI A.

Presenters: DREES T.; FEUCHTER D.; STARY T.; WINANDI A.

Session Classification: Facets of large Software Infrastructures

Track Classification: Education: project planning and project management