# Building sustainability and community in a small project: lessons from working on SaltProc

Oleksandr R. Yardas

Dr. Madicken Munk

University of Illinois Urbana Champaign

## Introduction

SaltProc is a Python package for simulating nuclear fuel reprocessing created as part of a dissertation project. It was originally built around an export-controlled software project. I took over as maintainer after the original author was unable to continue working on the project. The project had essentially no users, but I did not want to see SaltProc die so came up with a plan to make the project easier to use.

## What we did

- Set up GitHub Discussions to record design decisions and as a platform to interact with users
- Added support for an open-source alternative to Serpent2
- Overhauled the web hosted docpages with a guide on how to use the package
- Used automation tooling to perform rote tasks (releases, updating and deploying docpages, testing)
- Replaced a difficult-to-install dependency with friendlier alternatives

## Conclusion

After implementing the open source feature and creating the discussions page, SaltProc began to get more stars and we saw interaction from users on the discussion page.

## Open issues

SaltProc only has one recent contributor. In niche packages like SaltProc, how we get good users who become contributors?

Small open source projects reliant on non-open components may struggle to attract users. Adding support for open alternatives to these components will attract users.

**Take a picture** to check out SaltProc

```
{
    "$schema": "https://json-schema.org/draft/2020-12/schema",
    "$id": "https://github.com/arfc/saltproc",
    "title": "SaltProc input file schema",
    "type": "object",
    "properties": {
        "proc_input_file": {
            "description": "File containing processing system objects",
            "type": "string",
            "pattern": "^(.*)\\.json$"
        },
        "dot_input_file": {
            "description": "Graph file containing processing system structure"
            "type": "string",
            "pattern": "^(.*)\\.dot$"
        },
        "output_path": {
            "description": "Path output data storing folder",
            "type": "string",
            "pattern": "^(.\\/)*(.*)$",
            "default": "saltproc_runtime"
```