Open Source .Science MOSS IG Call

Software Gardening Almanack

Image credit: H. Zell

Agenda

1 Introduction
2 Challenges
3 Solutions

Image credit: Ciara Ní Riain

Introduction

Dave Bunten, M.Ed.
Research Software Engineer

University of Colorado
Anschutz Medical Campus

Software Engineering Team (link)
Department of Biomedical Informatics
School of Medicine
University of Colorado Anschutz

better
scientific
software

Better Scientific Software (BSSw)
2024 Fellow (link)

We have a problem.

"Concerns are growing about the **productivity** of the developers and users of scientific software, its **sustainability**, and the **trustworthiness** of the results that it produces."
Heroux et al. (2023)

"... the national **annual costs** of an inadequate infrastructure for software testing is estimated to range from **$22.2 to $59.5 billion**"
RTI (2002)

The cost of poor software quality was estimated to be around **$2.41 trillion** in 2022.
CISQ (2022)

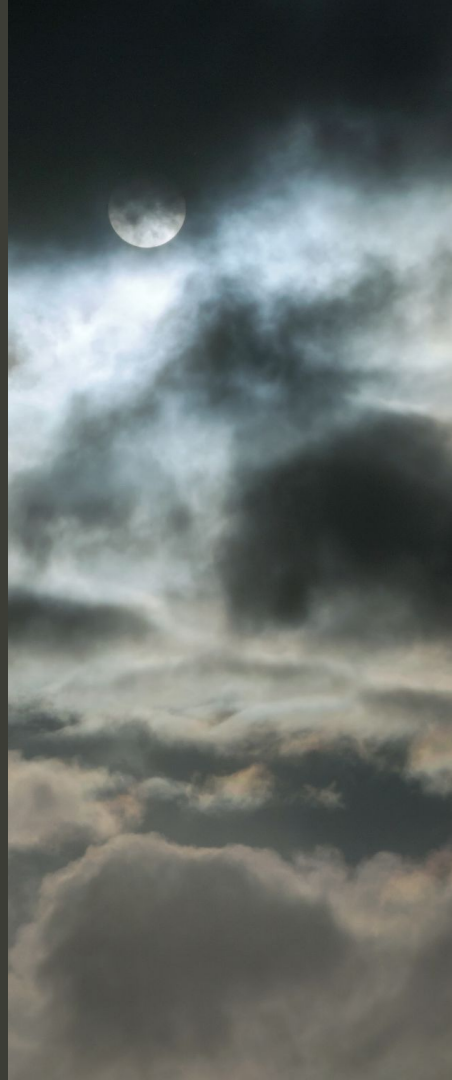


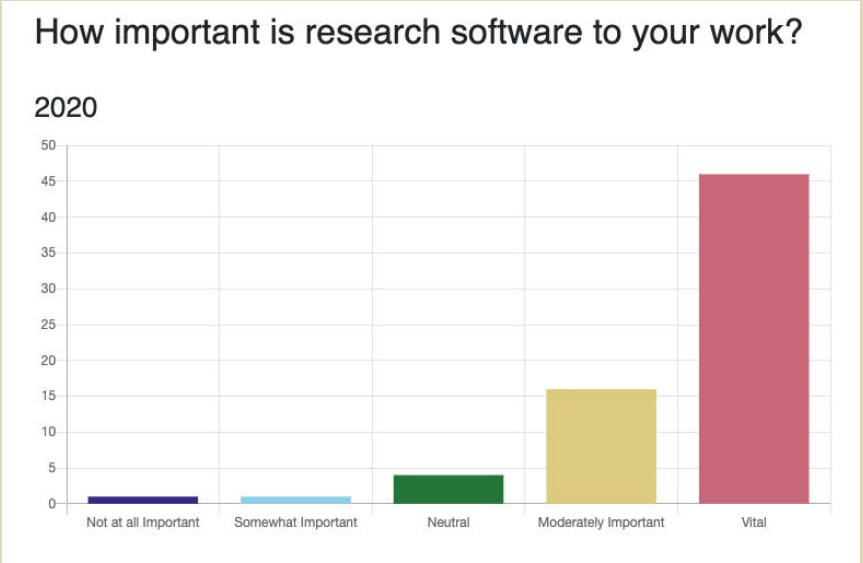
Image credit: W.carter

Why scientific software?

"Software has become an **essential part** of modern science, impacting discoveries, policy, and technological development."
Heroux et al. (2023)

"Biomedical software has become a **critical component** of nearly all biomedical research."
Afiaz et al. (2023)

"... most scholarly research relies on the same key resource: software."
US-RSE Association, IEEE Society (2023)



Sochat, V. (2021)

How much scientific software?

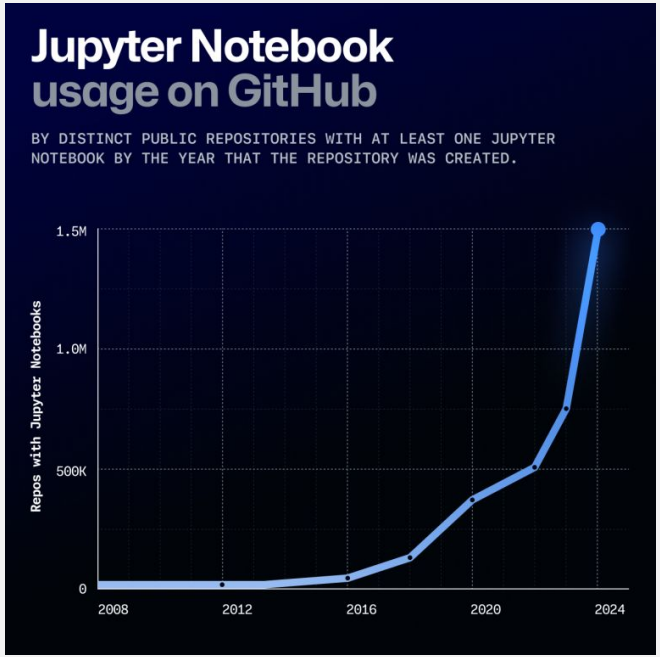


Image credit: GitHub Staff

~1.5 million repositories on GitHub created in 2024 contained at least one Jupyter notebook. GitHub Octoverse 2024

~152,888 results from Zenodo search for software records (Jan 2025). (search query)

~8,174 unique open source repositories with connections to DOE national laboratories. Schwartz et al. (2024)

Scale Challenges

We'd need **~367,283** software engineers each year to resolve these existing software challenges. There were **~10,000** research software engineers (RSE's) globally as of 2023. Cosden et al. (2023)

Median S.Eng. salaries: ***$162 thousand per year.*** (Glassdoor)

How could we resolve ***$59.5 billion per year*** in research Software needs (US)?

Cultivating outsized research software impact

There are orders of magnitude differences in productivity between software developers.
McConnel, Steve (2004)

Undergraduate and graduate programs may produce a better prepared workforce for Research Software Engineering in the future.
Goth et al. (2024)

"Without data, you're just another person with an opinion." - W. Edwards Deming

Image credit: Keith Evans

Existing measures don’t show the full picture

GitHub stars are sometimes fake and may be associated with bots or scams.
He et al. (2024)

Publication and citation metrics suffer from diminishing returns due to increased numbers, larger author or reference lists, and self-citations.
Fire & Guestrin (2019)

Could we measure scientific software for sustainability to mitigate costs from challenges?

Image credit: Antonio Tempesta

Software Linters

Linters perform **static analysis** of code to provide guidance on **bug avoidance** and **best practices** before software is delivered.

Historically, all software linters stem from a C debugger called “lint”. Johnson, Steven C. (1978)

Modern iterations include for example pylint (2001) and ruff (2022).

Pylint
Star your Python code!

Ruff

```
$ uv run ruff check
src/numbers/__init__.py:5:80: E501 Line too long (90 > 79)
Found 1 error.
```

Image credit (top to bottom and left to right): Frank C. Müller, Wikipedia, Astral docs

Linters are an *educational technology*

Image credit: Winslow Homer

Linters provide a scalable opportunity to teach developers how to improve their code as they create their work (before premature decay).

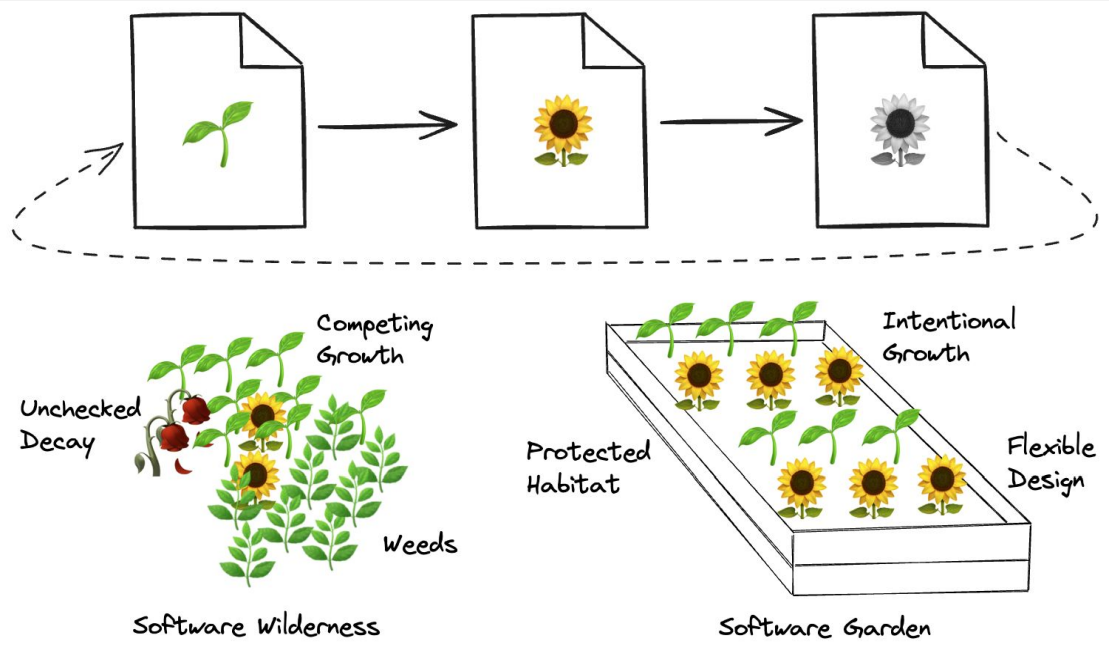
Creating linters often entails a community of practice which builds a curriculum that can be deployed through real-world “classrooms”.

Perhaps "24x engineers" aren’t born—they’re shaped by learning and empowered by a culture of teaching to mentor others.

Software Gardening

Image credit: Charles-François Daubigny

Time and software



Software undergoes changes over time, similar to a garden (or wilderness).

There are practices which help slow decay or recycle that energy into other growth.

Bunten & Way (2023)

The Almanack

The **Software Gardening Almanack** is an open-source handbook of applied guidance and tools for sustainable software development and maintenance.

The Almanack:

- Educates developers
- Analyzes software metrics
- Provides open measurements
- Empowers a culture of change

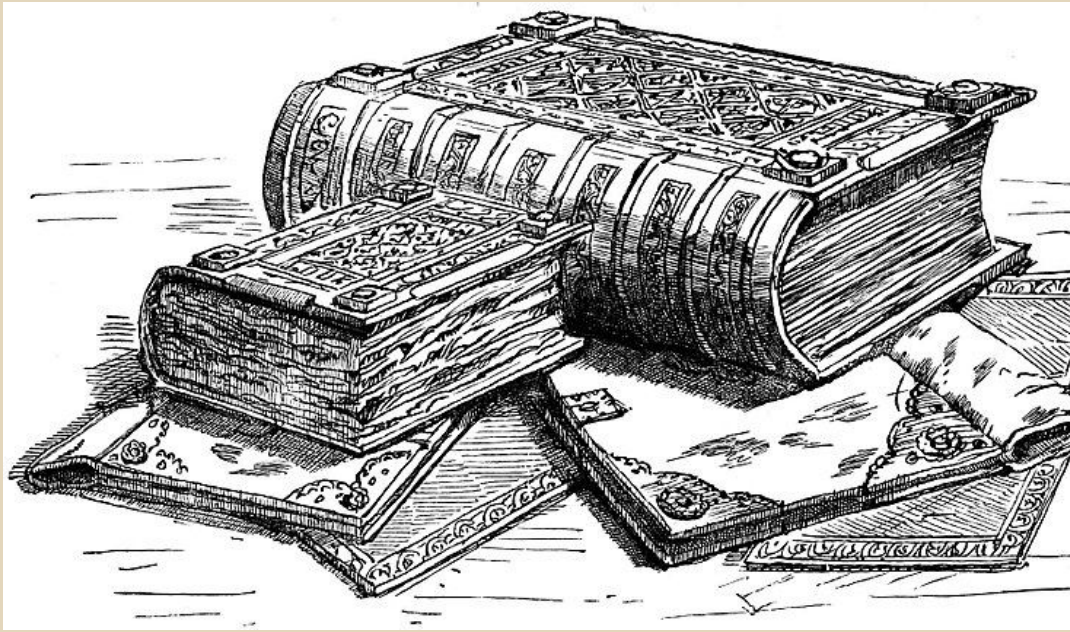
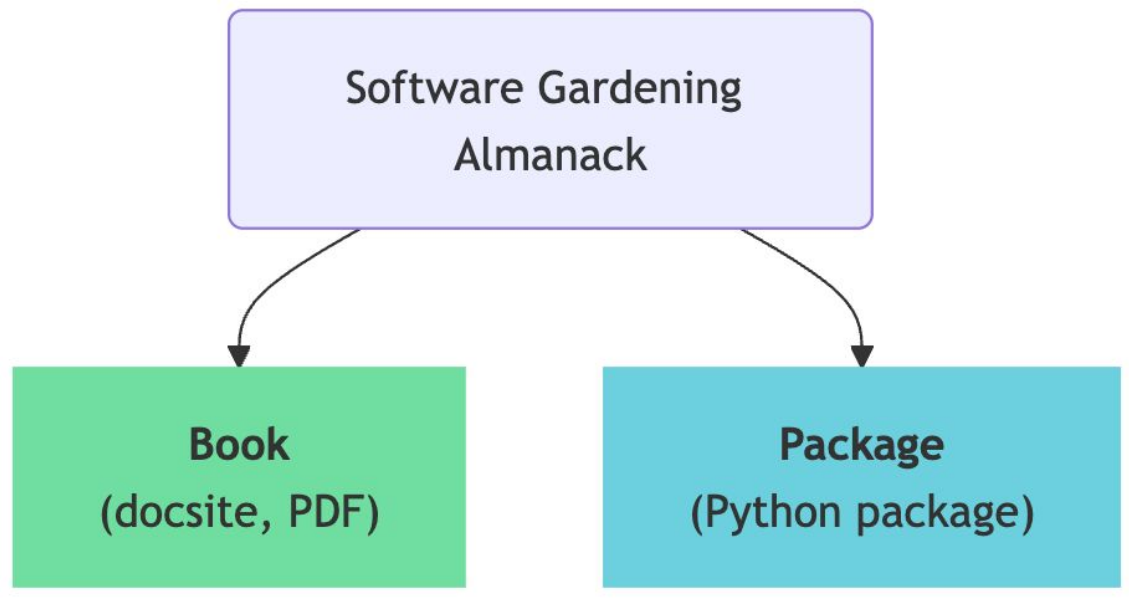


Image credit: Andrews, William

Almanack Components



The Almanack is composed of:

- **Book** (docsite and PDF): *Used for learning and documented analysis.*
- **Package** (PyPI): *Used for applied discovery of patterns within repositories (software gardens).*

Almanack Book

Search ⌘ + K

- Welcome
- Garden Lattice
- Software Forest
- Verdant Sundial
- Seed Bank
- Garden Circle

Welcome

Welcome to the Software Gardening Almanack, an open-source handbook of applied guidance and tools for sustainable software development and maintenance.

Inspiration

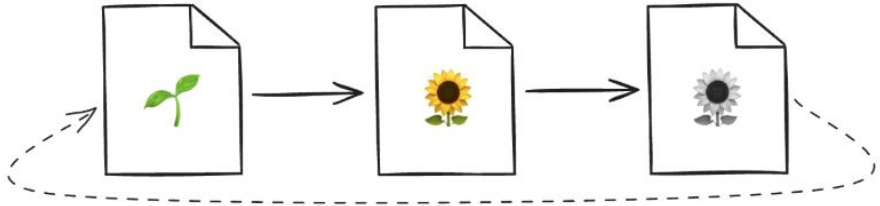


Fig. 1 Software is created, grows, decays over time.

Software experiences development cycles, which accumulate errors over time. However, these cycles are not well understood nor are they explicitly cultivated with the impacts of time in mind. Why does software grow quickly only to decay just as fast? How do software bugs seem to appear in unlikely scenarios?

These cycles follow patterns from life: software is created, grows, decays, and so on (sometimes in surprisingly unpredictable ways). Software is also connected within a complex system of dependencies (similar to the complex ecology of a garden). The Software Gardening Almanack posits we can observe software lifecycle patterns and complex relationships in order to build tools which sustain or maintain their development long-term.

> *"The 'planetary garden' is a means of considering ecology as the integration of humanity – the gardeners – into its smallest spaces. Its guiding philosophy is based on the principle of the 'garden in motion': do the most **for**, the minimum **against**."* - Gilles Clément

Contents

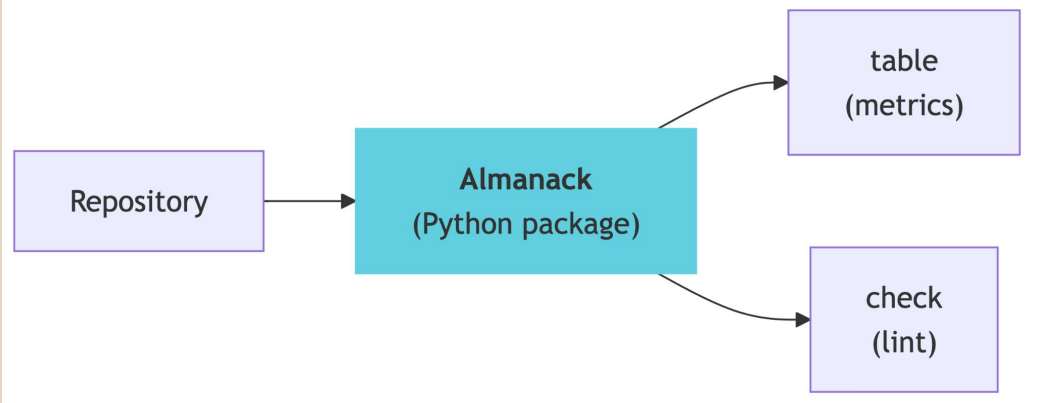
Almanack Package

Install the package from PyPI:
`pip install almanack`

Create a table of metrics:
`$ almanack table <repo path>`

Lint a repository for best practices:
`$ almanack check <repo path>`

[Example Google Colab notebook](#) for a quick demonstration.




```python
import almanack
import json
import pandas as pd

# gather the almanack table using the almanack repo as a reference
almanack_table = almanack.metrics.data.get_table("almanack")

# show the almanack table as a Pandas DataFrame
pd.DataFrame(almanack_table)
```

	name	id	result-type	description	result
0	repo-path	SGA-META-0001	str	Repository path (local directory).	/content/almanack
1	repo-commits	SGA-META-0002	int	Total number of commits for the repository.	119
2	repo-file-count	SGA-META-0003	int	Total number of files tracked within the repos...	105
3	repo-commit-time-range	SGA-META-0004	tuple	Starting commit and most recent commit for the...	(2024-03-05, 2024-11-21)
4	repo-days-of-development	SGA-META-0005	int	Integer representing the number of days of dev...	262
5	repo-commits-per-day	SGA-META-0006	float	Floating point number which represents the num...	0.454198

Almanack Roadmap

Build Collaborations

Put the Almanack in the hands of people who can benefit from or can help develop it!

CLI Linter

Release CLI-based linter tool for applied guidance.

Pre-commit hook

Release pre-commit hook for ease of implementation for CLI linter.

Reusable GitHub Action

Release GitHub Action to for use as part of CI/CD workflows.

Sustainability Score

Research how repository metrics influence sustainability culminating in a single score.

bioRxiv
THE PREPRINT SERVER FOR BIOLOGY

Integrate with bioRxiv

Create bioRxiv integration to evaluate software from pre-prints on-site.

MOSS and the Almanack

The Almanack could benefit from data gathered by MOSS.

MOSS could include data formulated by the Almanack (e.g. sustainability score).

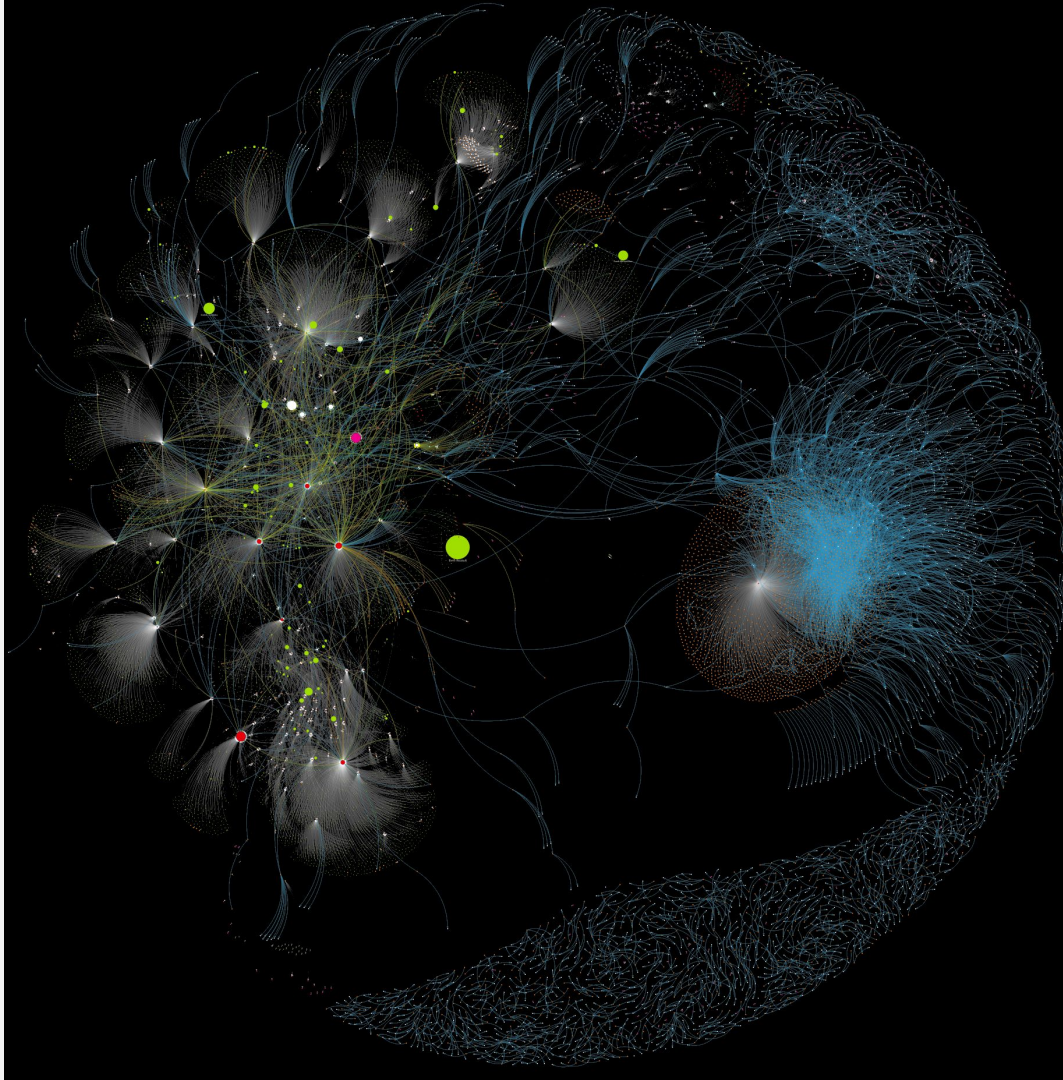
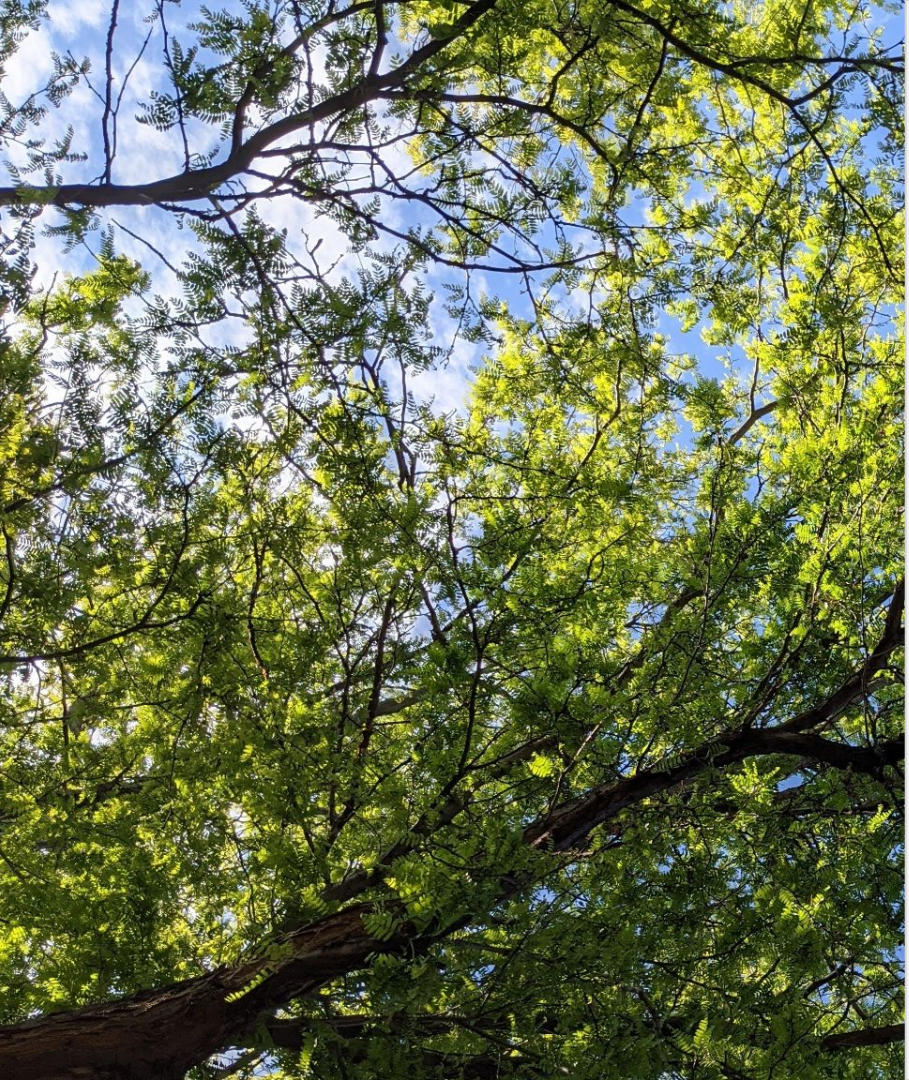


Image credit: Opensource.science



Let’s garden to thrive in a better future!

Software Gardening Almanack

https://github.com/software-gardening/almanack

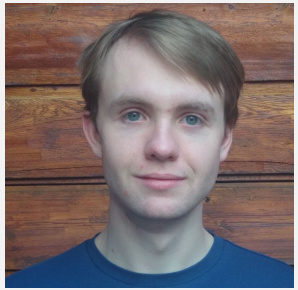
Acknowledgements

Gregory P. Way, PhD

Will Davidson

Faisal Alquaddoomi, PhD

Vince Rubinetti



Cameron Mattson

The Way Lab

Department of Biomedical Informatics
SCHOOL OF MEDICINE
UNIVERSITY OF COLORADO **ANSCHUTZ MEDICAL CAMPUS**

https://medschool.cuanschutz.edu/dbmi

better scientific software

https://bssw.io/

SUSTAINABLE HORIZONS INSTITUTE

https://shinstitute.org/

Thank you!

Questions / comments?

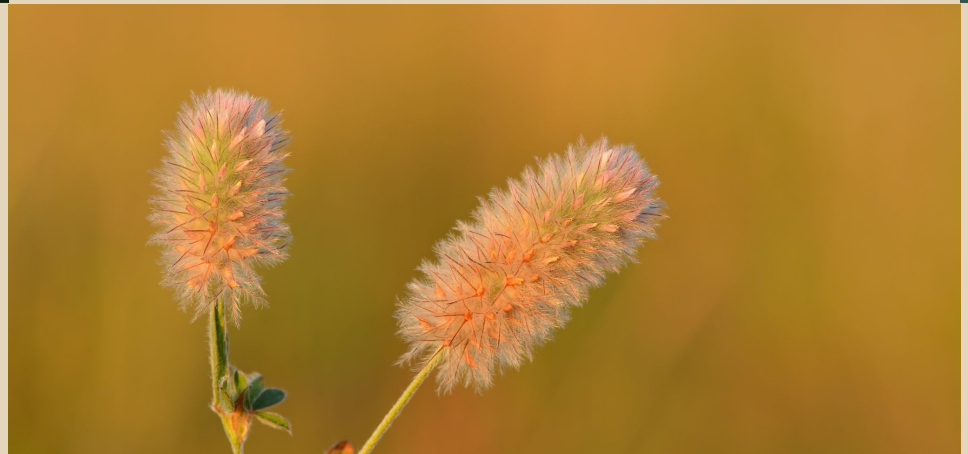


Image credit (left to right):
Joaquim Alves Gaspar,
Ivar Leidus,
Sam Oth

Software Gardening Almanack

Find us on GitHub!

Get in touch!
GitHub: @d33bs