The Software Gardening Almanack

Dave Bunten[1], Will Davidson[1], Faisal Alquaddoomi[1], Vincent Rubinetti[1], Gregory P. Way[1]

[1]*Department of Biomedical Informatics, University of Colorado Anschutz Medical Campus*

Scan for GitHub!

I. Fragile code & sustainability debt

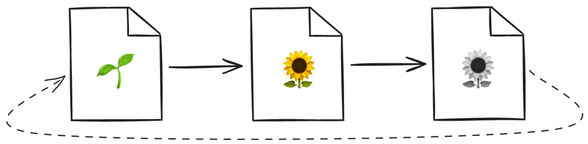


Figure 1: ***"Software grows and decays." Repositories accumulate complexity, missing docs, and other issues that erode maintainability over time. The Almanack treats software like a living ecosystem and provides measurements + checks to slow decay and guide care.***

Scientific software often becomes brittle: documentation gaps, weak tests, and ad-hoc practices combine with time to create **sustainability debt**. This debt hides defects, slows research, and undermines reproducibility. Existing guidance is scattered and qualitative. Teams need **measurable signals** and **actionable checks** that map to better long-term outcomes.

II. Handbook + Python package

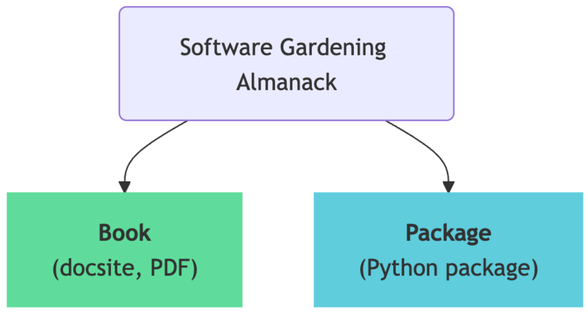


Figure 2: ***Two parts: a public handbook (education, examples, and rationale) and a Python package that generates metrics and runs checks on any repo. Integrate as CLI, pre-commit hook, or Python API to surface sustainability risks early.***

We present **The Software Gardening Almanack**, an open-source project with two complementary components:

- 📗 **Handbook (Jupyter Book):** teaching the "why" and "how"—concepts, examples, and tutorials. The handbook provides a portable foundation for learning sustainable practices.
- 🎻 **`almanack` Python package:** the "do"—compute metrics (including software information entropy), build reports, and run lint-style **sustainability checks** through Python API, CLI, or pre-commit CI.

Together, they help scientific developers quantify maintenance risk, prioritize fixes, and cultivate code that lasts. **The Almanack fills an educational technology gap** which endemic in scientific software development today.

III. Learning with the Almanack Handbook

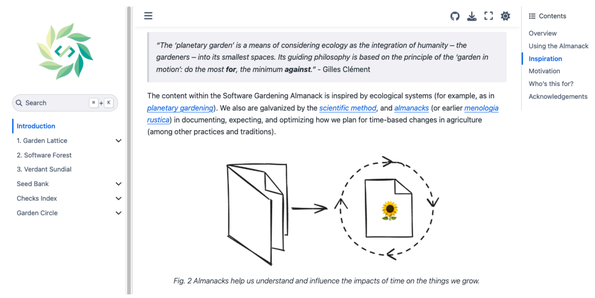
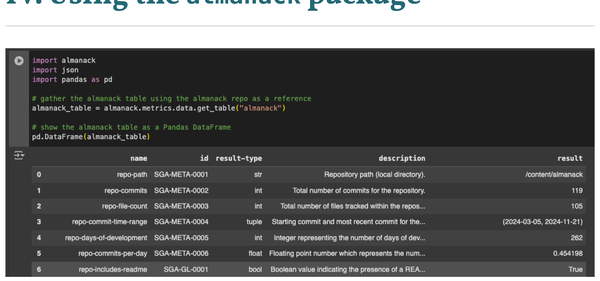


Figure 3: ***The Almanack handbook provides a learning avenue and reference manual for sustainable software development.***

Educational material is provided for reference and general training through a Jupyter Book for the Almanack. This content covers the philosophical foundations and references to specific guidance for various checks.

IV. Using the `almanack` package



```
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
6	repo-includes-readme	SGA-GL-0001	bool	Boolean value indicating the presence of a REA...	True

Figure 4: ***The Almanack package allows you to gather raw metrics for deep analysis or perform a lint-style check on repositories.***

Install with `pip`

```
# install from PyPI
pip install almanack
```

Use as a shell CLI

```
# run lint-style sustainability checks
almanack check https://github.com/an-org/your-repo

# gather all metrics about a project
almanack table path/to/repository
```

Pre-commit integration

```
# .pre-commit-config.yaml
# use the almanack directly through pre-commit
- repo: https://github.com/software-gardening/
almanack
  rev: v0.1.1
  hooks:
    - id: almanack-check
```

V. What the metrics capture

- **Repository hygiene:** README, license, citation, contribution guide, code of conduct, docs location.
- **Process health:** CI configuration, build status, test presence, coverage parsing (e.g., Python).
- **Community signals:** contributors, tags/releases, issue tracker availability.
- **Evolution/complexity: information entropy** at file and repo levels to highlight hotspots and decay risk.
- **Scoring & gating:** boolean checks mapped to maintenance direction (positive/negative) plus an overall **Almanack score**.

VI. Visual analytics & examples

The **Seed Bank** demonstrates how to reproduce and explore measures (e.g., normalized Shannon entropy) across thousands of research repositories, relating **time**, **complexity**, and **engagement**.

VII. Typical workflows in a lab

1. **Survey repositories** to establish a baseline metrics table and identify immediate fixes (missing license, broken badge, etc.).
2. **Add guardrails** with pre-commit/CI so regressions are caught early (e.g., missing citation file).
3. **Plan maintenance** using entropy hotspots and failed-check rationales to guide refactoring and documentation.

VIII. Real-world value for scientists

- **Reproducibility & trust:** standard checks and reports make expectations visible for collaborators and reviewers.
- **Onboarding faster:** new team members see structure (docs/tests/CI) and can contribute earlier.
- **Longevity:** entropy and process signals help choose where to invest limited maintenance time.
- **Automation-ready:** outputs integrate cleanly with dashboards, CI logs, and PR comments.

IX. Acknowledgements

We thank contributors and supporters of the Almanack effort, including the **Better Scientific Software (BSSw) Fellowship Program**, **U.S. Department of Energy (DOE)**, **National Nuclear Security Administration (NNSA)**, and the **National Science Foundation (NSF, award 2327079)**; and collaborators across the University of Colorado Anschutz Medical Campus.

- Project: https://github.com/software-gardening/almanack
- Handbook: https://software-gardening.github.io/almanack
- PyPI package: https://pypi.org/project/almanack/