Python Ireland

oomnitza

ENTERPRISE TECHNOLOGY MANAGEMENT

A Tale of two Kitchens

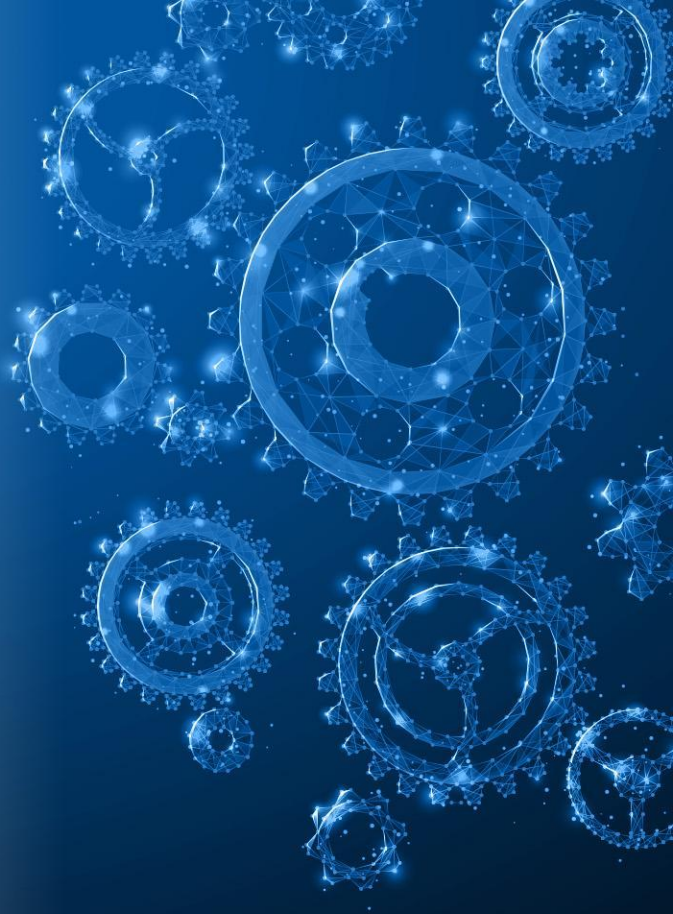
Hypermodernizing your Codebase

Christian Ledermann

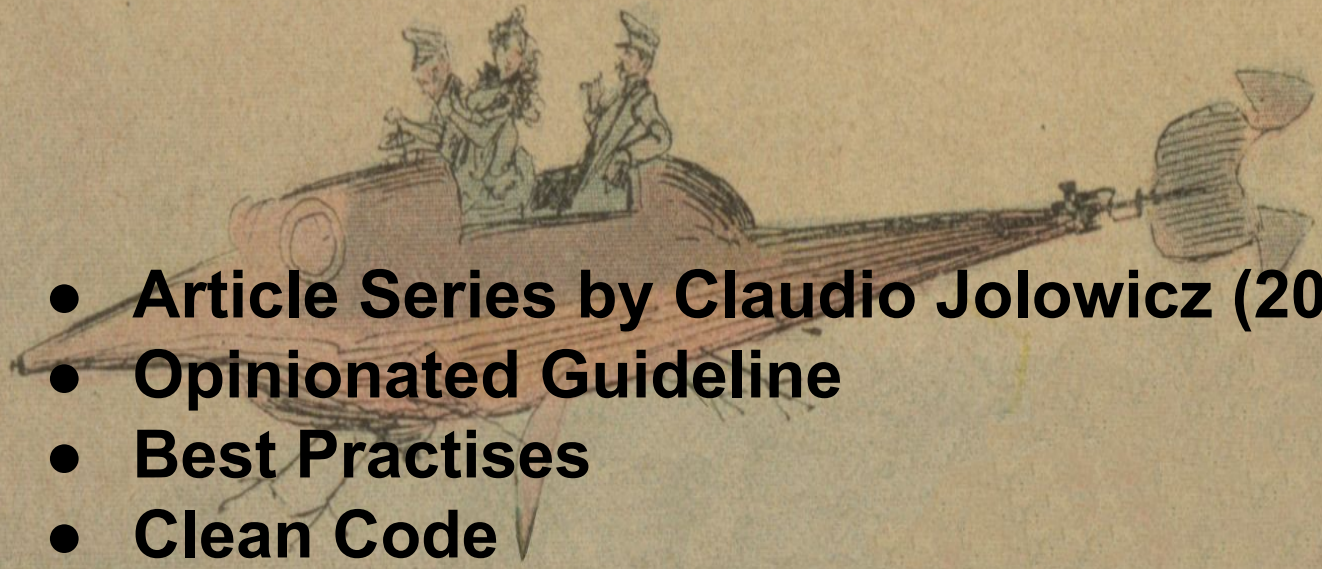
PyCon Ireland Limerick 25/03/2023

Overview

- What is Hypermodern Python
- Why Hypermodern Python
- How to hypermodernize your legacy Code



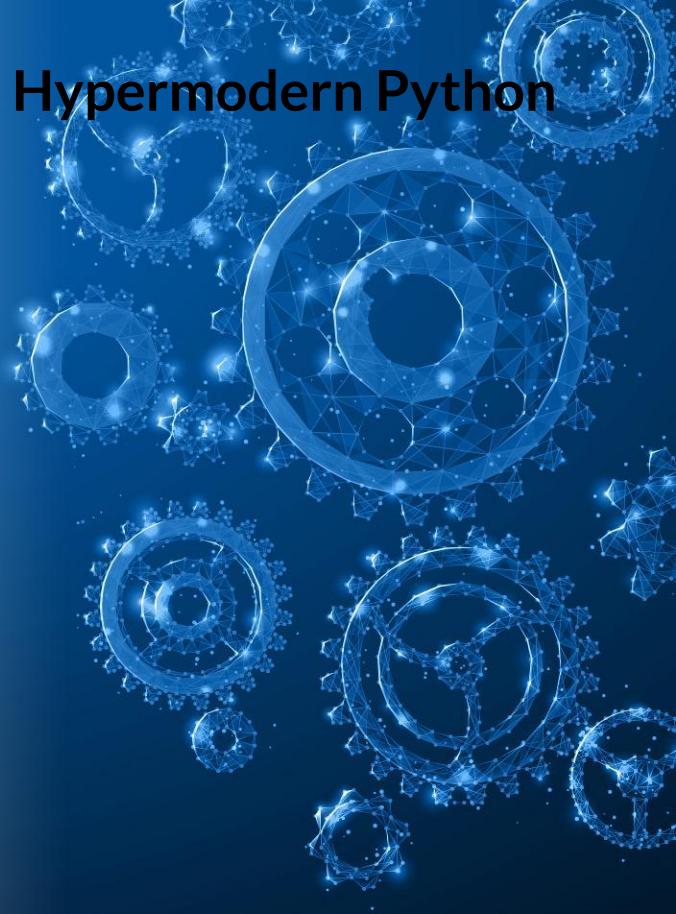
Hypermodern Python



- Article Series by Claudio Jolowicz (2020)
- Opinionated Guideline
- Best Practises
- Clean Code

Hypermodern Python

- Packaging and dependency management with Poetry
- Test automation with Nox
- Linting with pre-commit and Flake8
- Continuous integration with GitHub Actions
- Documentation with Sphinx, MyST, and Read the Docs using the furo theme
- Automated uploads to PyPI and TestPyPI
- Automated release notes with Release Drafter
- Automated dependency updates with Dependabot
- Code formatting with Black and Prettier
- Import sorting with isort
- Testing with pytest
- Code coverage with Coverage.py
- Coverage reporting with Codecov
- Command-line interface with Click
- Static type-checking with mypy
- Runtime type-checking with Typeguard
- Automated Python syntax upgrades with pyupgrade
- Security audit with Bandit and Safety
- Check documentation examples with xdoctest
- Generate API documentation with autodoc and napoleon
- Generate command-line reference with sphinx-click
- Manage project labels with GitHub Labeler



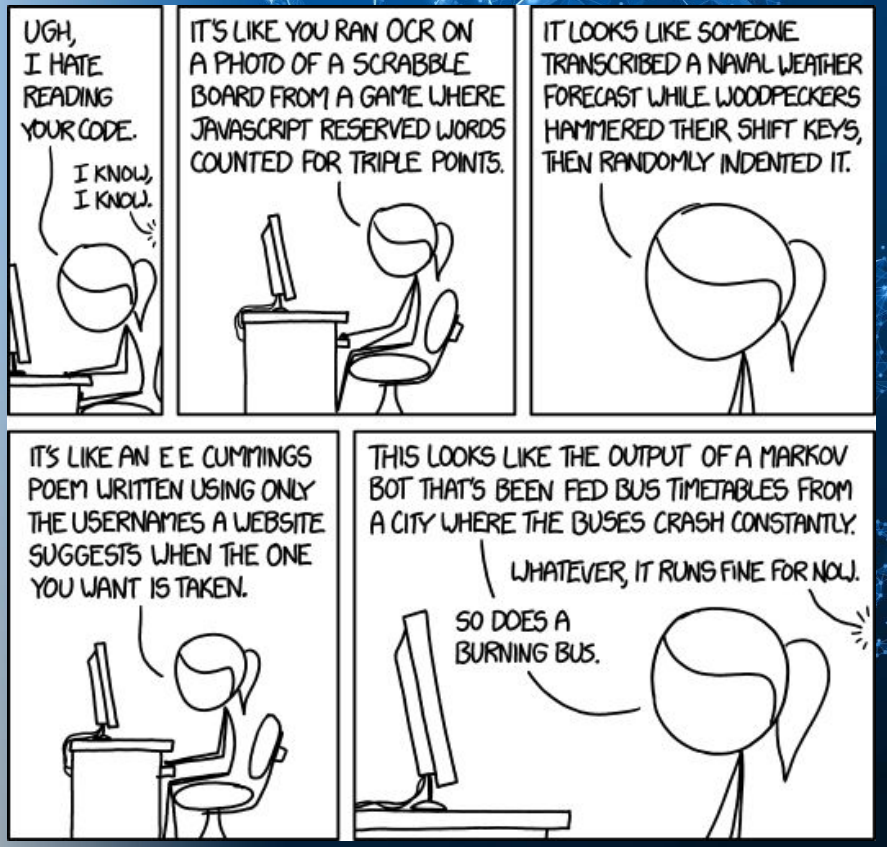
A Tale of Two Kitchens

Which Kitchen?

- Security
- Health and Safety
- Deliver Fast
- Deliver High Quality
- Job Satisfaction
- Professional Growth

Why Clean up your Codebase

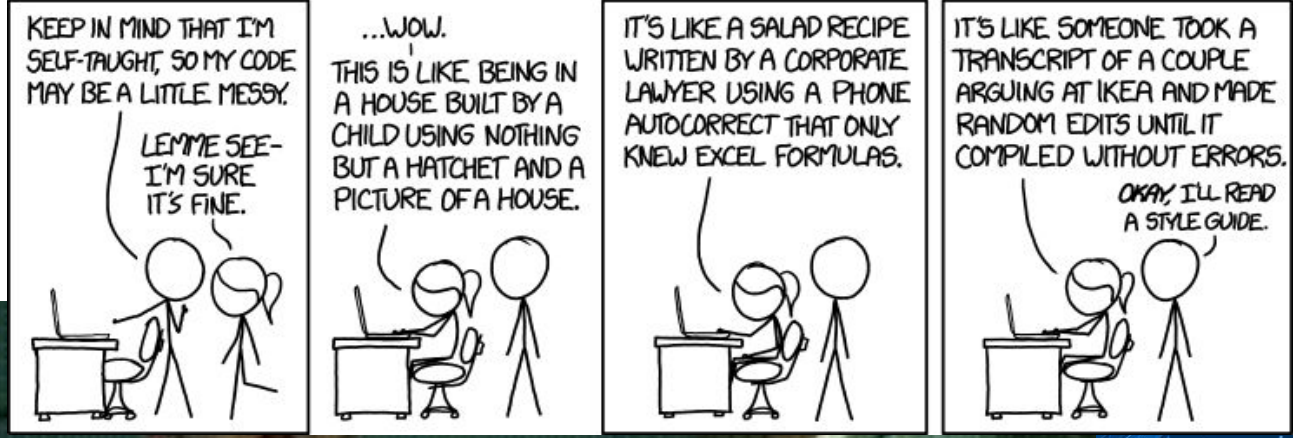
- Minimize Context switches
- LEAN: Eliminate Waste
- Broken Windows
- Boy Scout Rule
- Improve quality



The most important thing I have done as a programmer in recent years is to aggressively pursue static code analysis. Even more valuable than the hundreds of serious bugs I have prevented with it, is the change in mindset about the way I view software reliability and code quality.

-- John Carmack

Style matters




Start Small
Start Simple
Start Now
omnitza

Getting Started - Format Matters

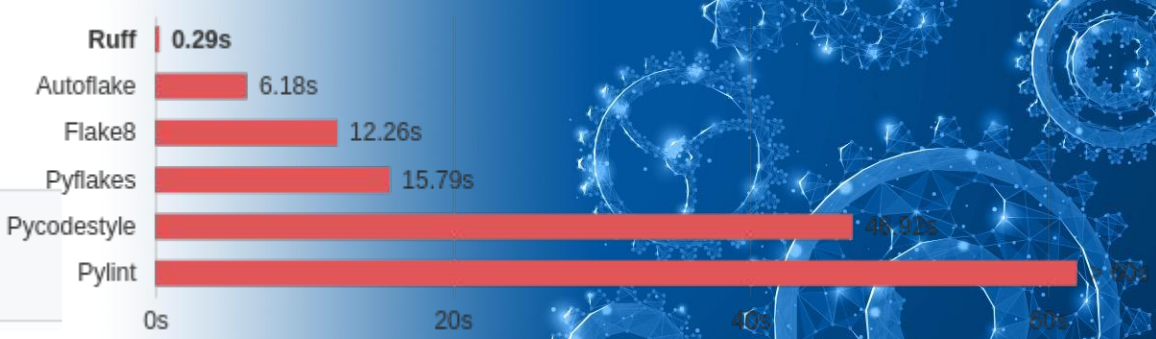
- pre-commit
- isort, absolufy-imports, unimport, remove-star
- black
- ~~flake8 + Plugins~~

Ruff

pypi v0.0.259

Initial commit for linter prototype

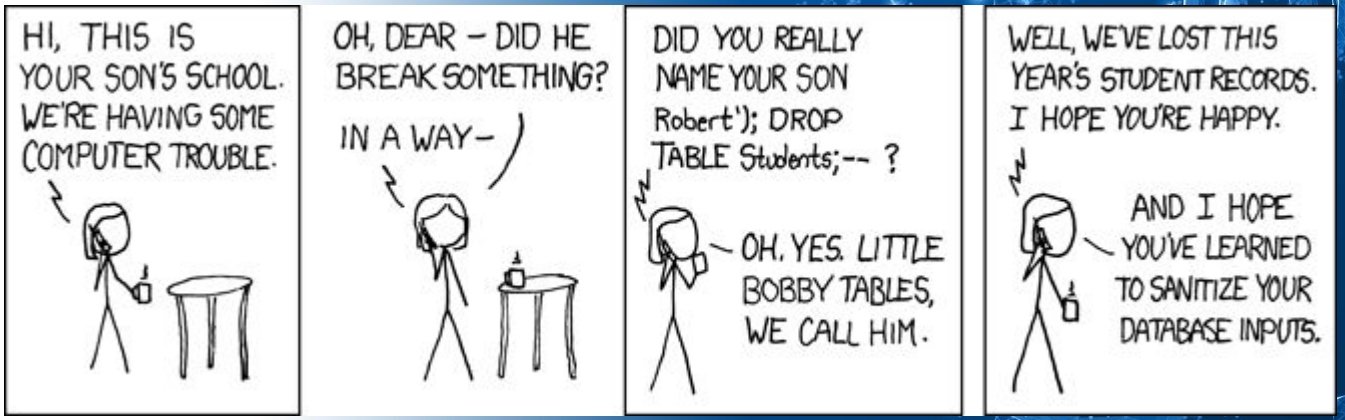
charliermarsh committed on Aug 9, 2022

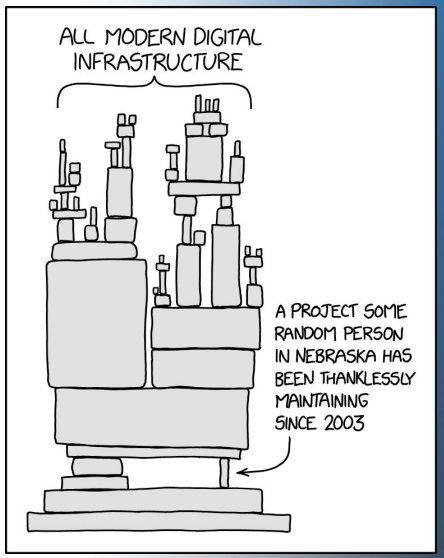


- ⚡ 10-100x faster than existing linters
- 🐍 Installable via `pip`
- 🛠️ `pyproject.toml` support
- 🤝 Python 3.11 compatibility
- 📦 Built-in caching, to avoid re-analyzing unchanged files
- 🔧 Autofix support, for automatic error correction (e.g., automatically remove unused imports)
- 📏 Over 500 built-in rules
- ⚖️ Near-parity with the built-in Flake8 rule set
- 🔌 Native re-implementations of dozens of Flake8 plugins, like flake8-bugbear
- ⌨️ First-party editor integrations for VS Code and more
- 🌎 Monorepo-friendly, with hierarchical and cascading configuration

Security

- ~~BugBear~~
- ~~Bandit~~
- Safety
- Dependabot
- Pysa





You improve what you measure

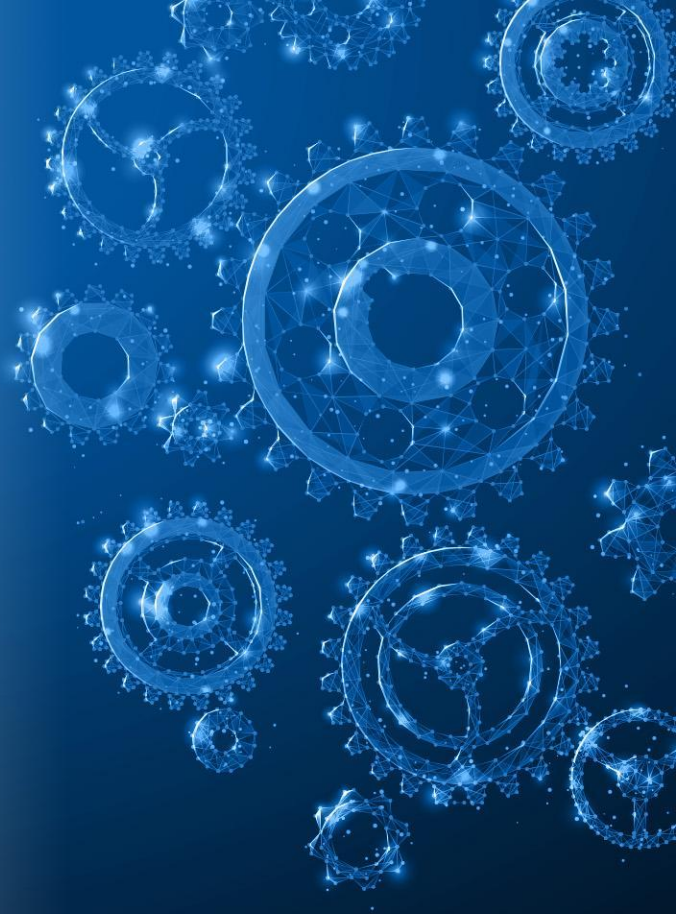
- Tests
  - Coverage
  - diff-cover
- Complexity
  - McCabe, Radon, Xenon
  - Lizard
  - Cognitive Complexity

Typing

- MyPy (pytype, pyre, pyright)
- MonkeyType
- pyre infer, pytype, pyright
- typeguard (for tests)

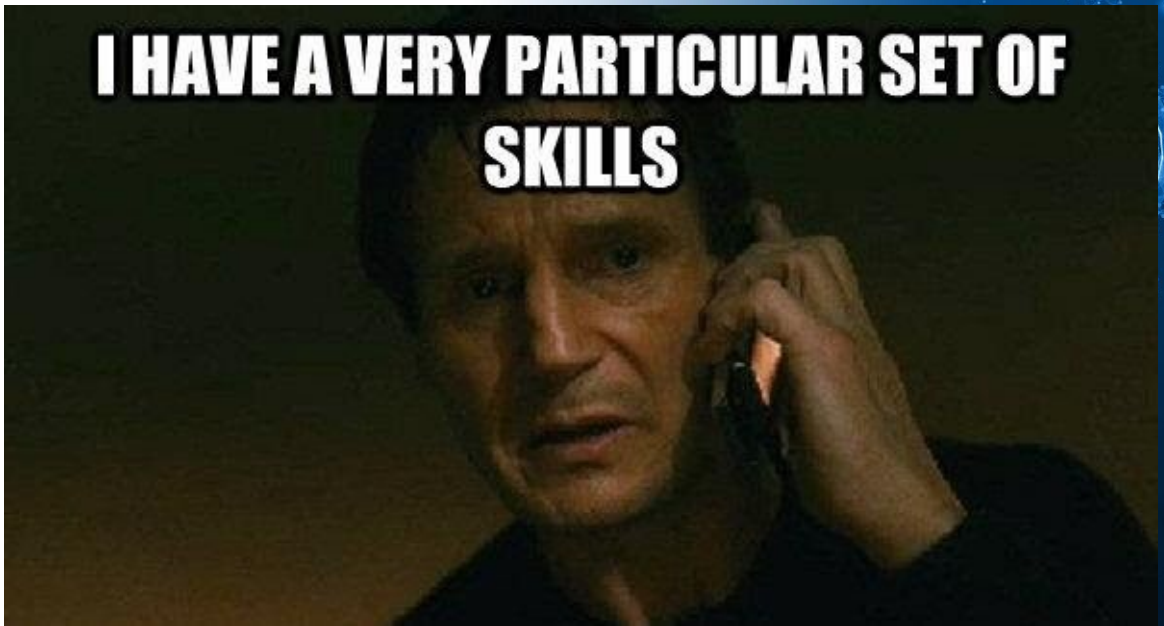
Tests need Love 💓 Too

- Teyit for unit tests
- pytestify
- unittest2pytest



Testing (continued)

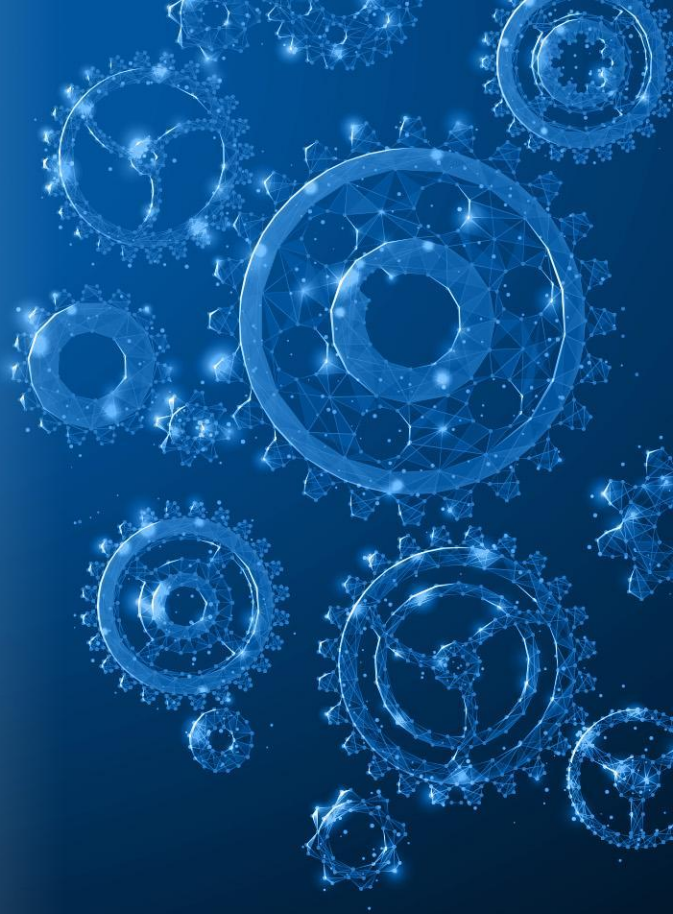
- Hypothesis and Ghostwriter
- Schemathesis for web APIs



Testing (continued)

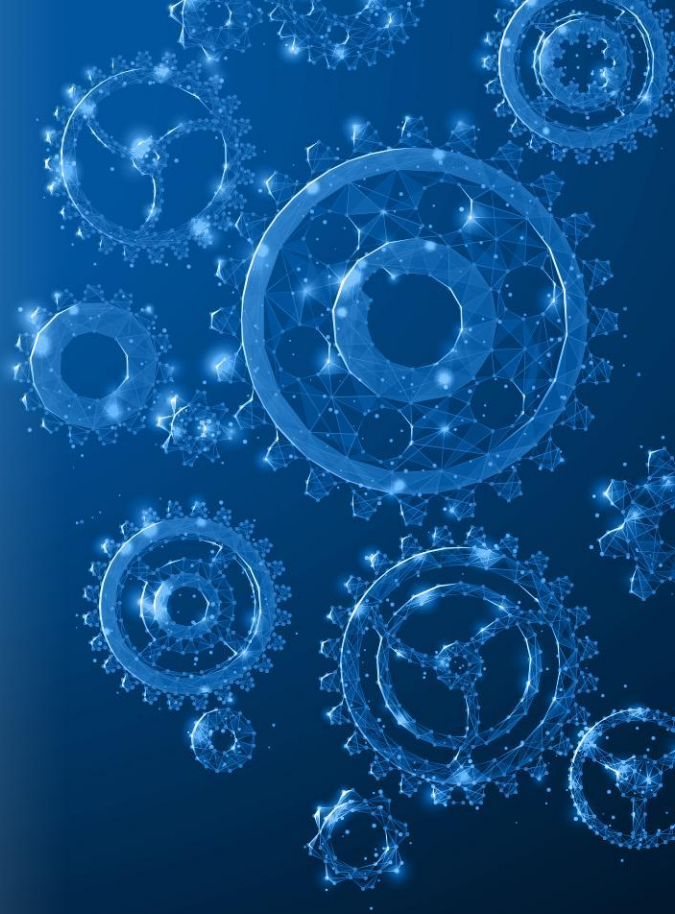
Quis custodiet ipsos custodes?
Who is watching those watchmen?

- MutMut Mutation Testing



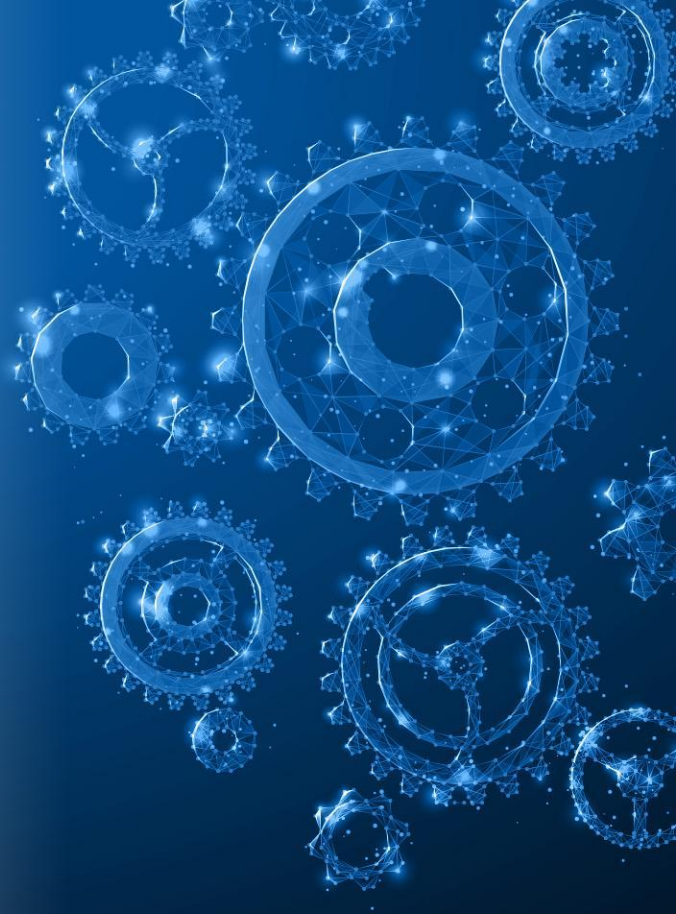
Upgrade

- ~~Pyupgrade~~
- ~~flynt~~
- Django-upgrade
- Django-codemod



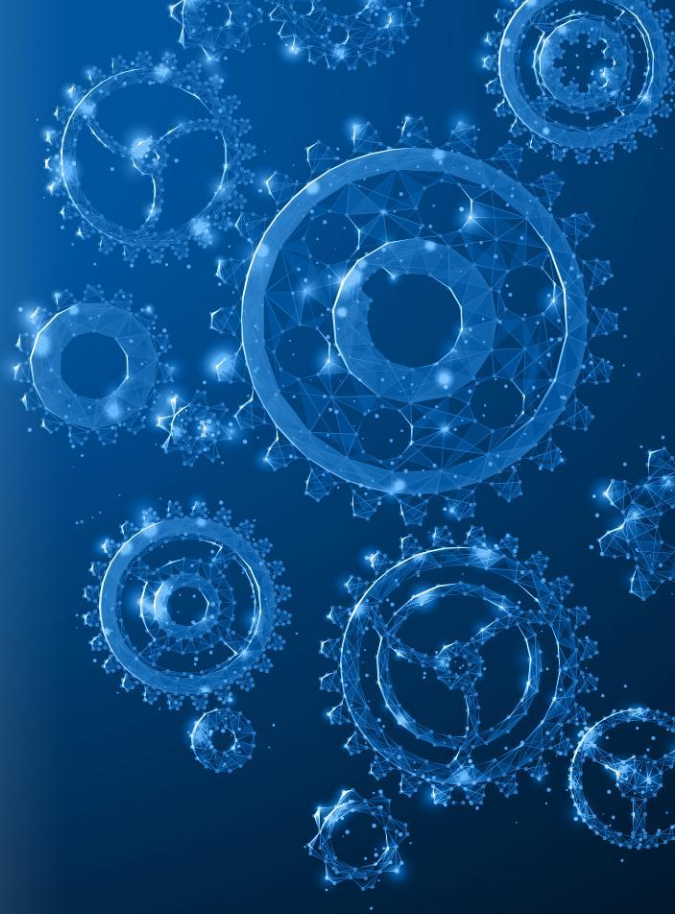
Roll our Own

- libCST
- …



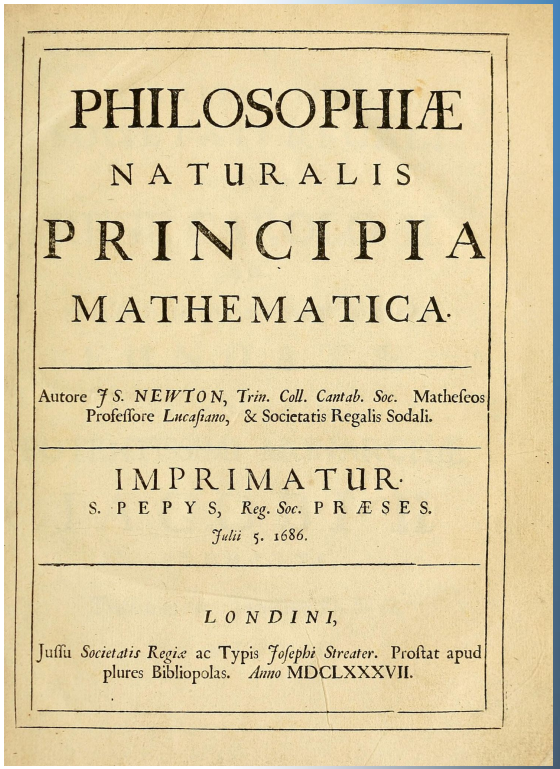
As A Service

- Sourcery
- SonarCloud
- Metabob
- Coverage.io



Where To Go From Here?

- DRY
- WET
- SOLID
- CUPID



PHILOSOPHIÆ
NATURALIS
PRINCIPIA
MATHEMATICA.

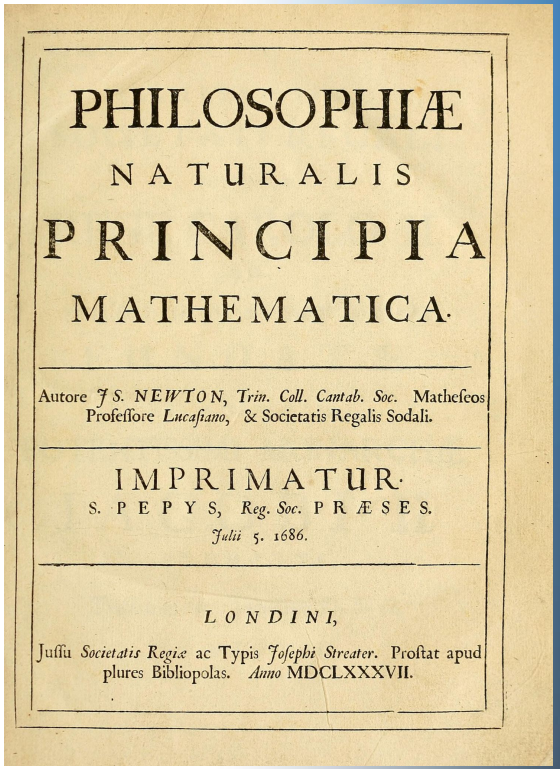
Autore *J S. NEWTON, Trin. Coll. Cantab. Soc.* Matheſeos Profeſſore *Lucaſiano*, & Societatis Regalis Sodali.

IMPRIMATUR.
S. PEPYS, *Reg. Soc.* PRÆSES.
Julii 5. 1686.

LONDINI,
Juſſu *Societatis Regiæ* ac Typis *Joſephi Streater.* Proſtat apud plures Bibliopolas. *Anno* MDCLXXXVII.

SOLID

- **S**ingle Responsibility
- **O**pen-Closed
- **L**iskov Substitution
- **I**nterface Segregation
- **D**ependency Inversion



PHILOSOPHIÆ
NATURALIS
PRINCIPIA
MATHEMATICA.

Autore *J S. NEWTON, Trin. Coll. Cantab. Soc.* Matheſeos Profeſſore *Lucaſiano*, & Societatis Regalis Sodali.

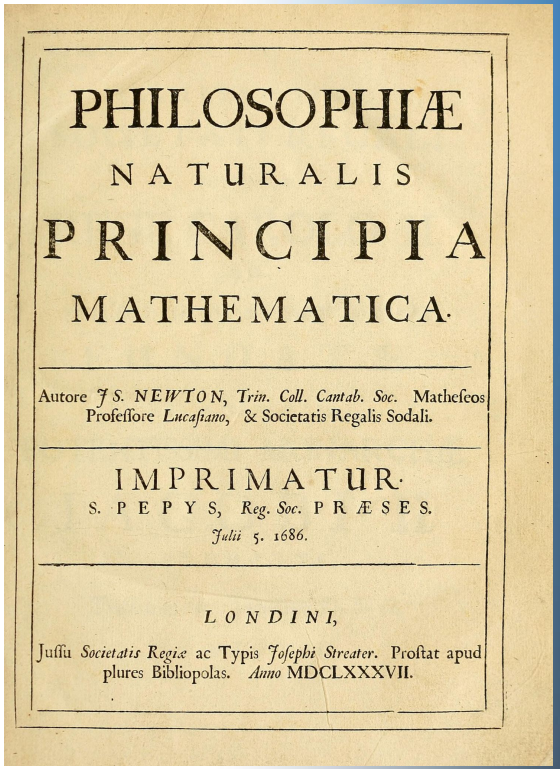
IMPRIMATUR.
S. PEPYS, *Reg. Soc.* PRÆSES.
Julii 5. 1686.

LONDINI,

Juſſu *Societatis Regiæ* ac Typis *Joſephi Streater.* Proſtat apud plures Bibliopolas. *Anno* MDCLXXXVII.

CUPID

- **Composable**: plays well with others
- **Unix philosophy**: does one thing well
- **Predictable**: does what you expect
- **Idiomatic**: feels natural
- **Domain-based**: the solution domain models the problem domain in language and structure



PHILOSOPHIÆ
NATURALIS
PRINCIPIA
MATHEMATICA.

Autore *J S. NEWTON, Trin. Coll. Cantab. Soc.* Matheseos Professore *Lucasiano*, & Societatis Regalis Sodali.

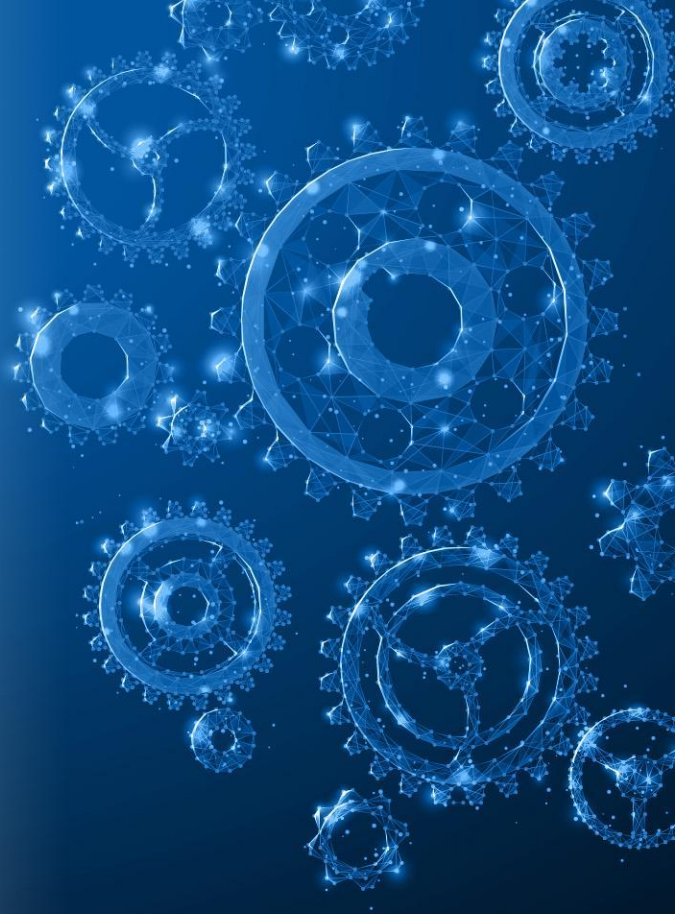
IMPRIMATUR.
S. PEPYS, *Reg. Soc.* PRÆSES.
Julii 5. 1686.

LONDINI,

Jussu *Societatis Regiæ* ac Typis *Josephi Streater.* Prostat apud plures Bibliopolas. *Anno* MDCLXXXVII.

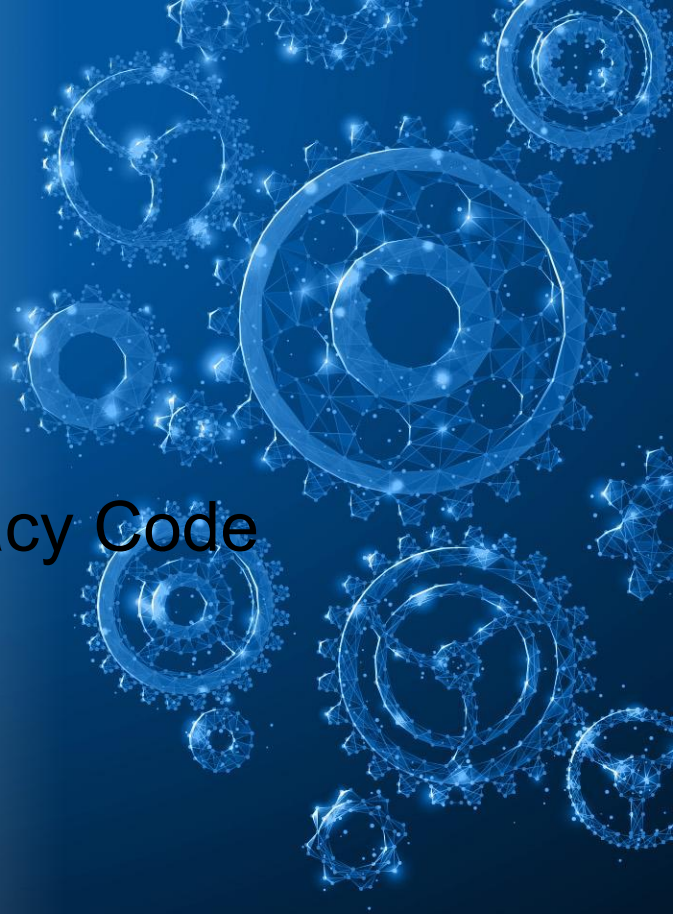
Means to Achieve an End

- Maintainability
- Extensibility
- Modularity



Perfect is the Opposite of Done

Summary



- What is Hypermodern Python
- Why Hypermodern Python
- How to hypermodernize your Legacy Code
- Next steps

Done is Better Than Perfect

THANK YOU

github.com/cleder/ep2022