

# Behind the Scenes of tox

Rewriting a Tool with more than ~~40~~ 12 Million  
Monthly Downloads

2023-11-05, Ubuntu Summit, Riga

Jürgen Gmach



Canonical



# Things You Should Never Do, Part I



“the single worst strategic mistake”



“They decided to rewrite  
the code from scratch.”

# All about me

## **Jürgen Gmach**

Senior Software Engineer, Canonical

@jugmac00 / @jugmac00@fosstodon.org

<https://jugmac00.github.io>

Maintainer of tox

PSF Fellow

Bavarian Forest, Germany



Canonical

# All about me

## Jürgen Gmach

Senior Software Engineer, Canonical

@jugmac00 / @jugmac00@fosstodon.org

<https://jugmac00.github.io/hiring>



hiring

Maintainer of tox

PSF Fellow

Bavarian Forest, Germany



Canonical



tox

# Testing a library or an app

## manual way

- create a virtual environment
- build package
- install the package
- install test dependencies, e.g. pytest, coverage
- run tests

And probably more, e.g. run linters, build documentation...



# Testing a library or an app

the tox way

```
● ● ●  
  
# tox.ini  
  
[tox]  
envlist = py38, py39, py310, py311, py312  
  
[testenv]  
deps = pytest > 8  
commands = pytest tests  
  
# `pipx install tox` and run it  
tox
```

# Initial tox release by Holger Krekel



2010-07-02

# Bernát Gábor joined the tox team



2017-06

# Soon he noticed...

tox was written with the following assumptions

It is used ...

- to create a Python environment
- on your local system
- to be configured via a tox.ini file
- to build source distributions
- to evaluate configuration eagerly

# Bernát Gábor starts the first rewrite attempt



2017-12-21



# Holger Krekel's last contribution



2018-03-24

# Bernát noticed a pattern in the bug reports for tox



# Bernát Gábor became a maintainer of virtualenv





# Bernát Gábor started rewriting virtualenv



I asked a fellow Zope maintainer how to test a package both with Python 2 and 3



2018-12-15

# Release of virtualenv

Lessons from the Trenches: rewriting and re-releasing virtualenv

<https://www.youtube.com/watch?v=l9A0a8qZgOs>



2020-02-10

# Bernát announced the tox 2nd ed. rewrite on Twitter



2020-02-17

# First commit for the tox rewrite



2020-03-29

# Bernát started streaming the rewrite on Twitch



# I started answering tox questions on StackOverflow



2020-10-05

# “Tool confusion - pythonic code with flake8, tox and pre-commit”

Python Ireland

<https://www.youtube.com/watch?v=9ZkjNJpUSLY>



2020-10-14



# I started contributing to discussions and github issues



2020-11-09

# Bernát announced tox4 alpha 2



2021-01-08

# Test tox4 alpha2 against 300 Zope repositories



22%

success rate

22%

success rate

“a 22% success here, not that great news 😊 but on the plus side seems if we fix zope we should be in a decent place”

Bernát Gábor

# Feedback on Twitter

Bernát: “From plugins point of view version 4 will be totally breaking.”

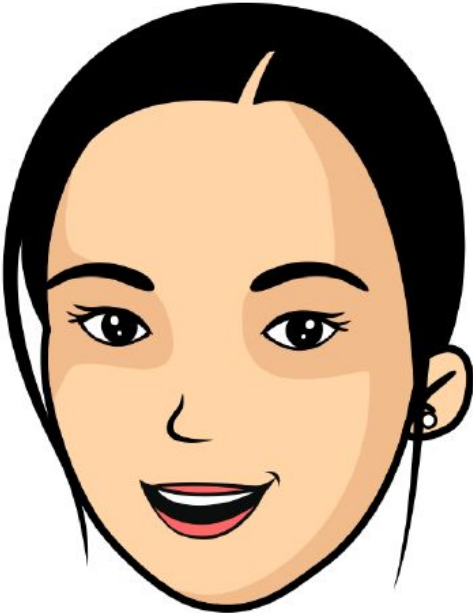
Oliver: “Are the people with the torches and pitchforks at the gates already?”



2021-03-01

# Feedback on Github

Cheuk Ting Ho: "I love tox"



2021-03-15

# First “serious” pull request



2021-03-20



# Bernát announced me as a tox maintainer



2021-04-05

# “Testing the tox4 Pre-Release at Plone-/Zope-Scale”

World Plone Day



2021-04-17

# Bernát announced tox4 at PyCon US via lightning talk



2021-05-15



# Life happened



# Life happened

- Bernát moved to the US



# Life happened

- Bernát moved to the US
- I built a house



# Life happened

- Bernát moved to the US
- I built a house
- I joined Canonical



# Torches and pitchforks



<https://pixabay.com/de/photos/m%c3%a4nner-appenzeller-brauchtum-394235/>



# Just release it

“That multi-year rewrite seems to never reach GA. If it was a business it would have being bankrupt one long time ago.”



2022-04-22

# Just release it

“That multi-year rewrite seems to never reach GA. If it was a business it would have being bankrupt one long time ago.”



2022-04-22

“Such a major release will ultimately break a lot of people, and we need maintainer availability to help those users and address their concerns.” Bernát Gábor

# Dependency with "vulnerable" version of pylib

a drama in two acts



2022-11-01

# Dependency with "vulnerable" version of pylib

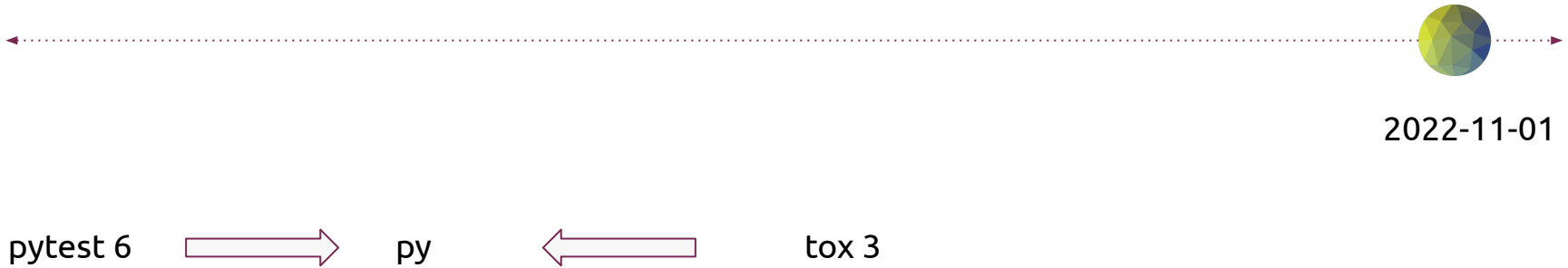
1. tox breaks CI



  
2022-11-01

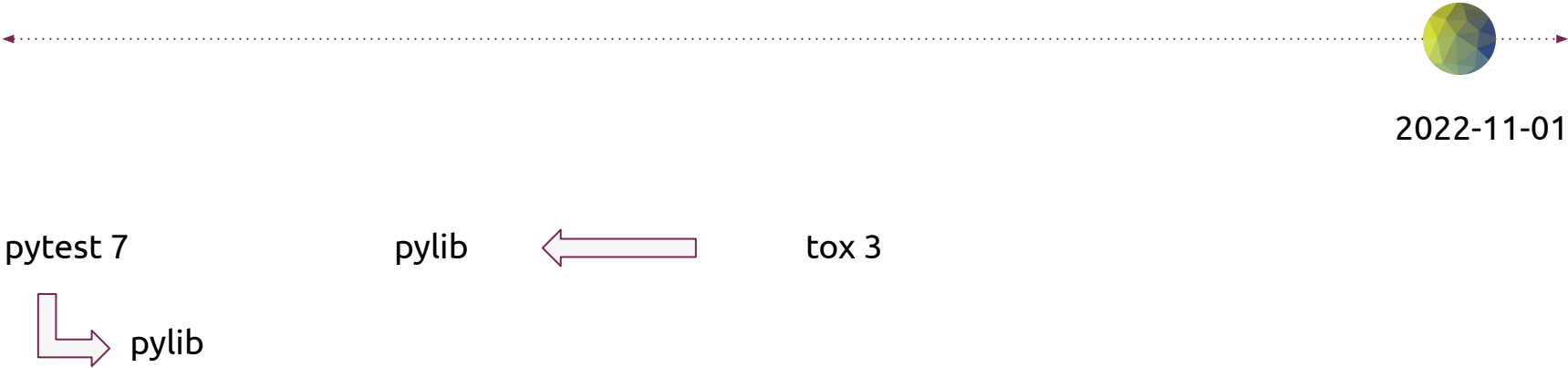
# Dependency with "vulnerable" version of pylib

## 2. tox gets broken



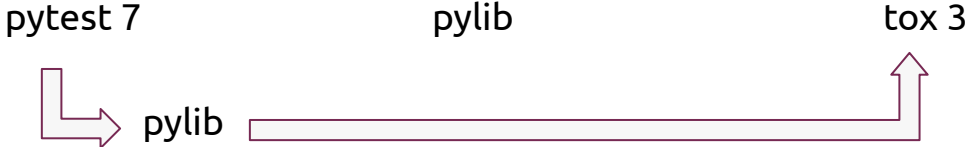
# Dependency with "vulnerable" version of pylib

## 2. tox gets broken



# Dependency with "vulnerable" version of pylib

## 2. tox gets broken



# Sorin to the rescue

Sorin Sbârnea announced to be able to work a full week on tox4.



2022-11-22



# Release

Only three years later!



2022-12-07

# Post-release different things happened

- lots of regressions
- lots of complaints
- lots of contributions from ~50 contributors
- lots of praise

# Hyrum's law

With a sufficient number of users of an API,  
it does not matter what you promise in the contract:  
all observable behaviors of your system  
will be depended on by somebody.

# I am not your supplier

<https://www.softwaremaxims.com/blog/not-a-supplier>

<https://testandcode.com/196>

# Highlights

- configurable via tox.ini, cli and environment variables
- faster
- easily extensible
- built-in wheel support
- support for grouping aka labels
- evaluate configuration lazily

<https://tox.wiki/en/latest/faq.html#new-features-in-tox-4>

# Abstractions

- config -> configuration management
- execute -> command execution inside environments
- journal -> result recording of a tox session
- plugin -> plugin management
- session -> command management
- tox\_env -> tox environment management

=> evolved, not right from the beginning

# What we did well

- automated test suite
- announcements on various channels and calls to action to test pre-releases
- stick around after release
- keep old documentation
- keep releasing critical updates to the tox 3 branch, though it is officially unmaintained

What we could have done better



# What we could have done better

nothing

# What we could have done better

nothing - given the resources

# Lessons learned

- have a plan
- be ready to re-evaluate everything from the ground up
- test a lot
- be available post-release
- prepare for bugfixes in the coming months after the release
- be nice, but set boundaries

And ...

# Lessons learned

... legends are not always right.



[https://commons.wikimedia.org/wiki/File:Joel\\_spolsky\\_on\\_20\\_sept\\_2007.jpg](https://commons.wikimedia.org/wiki/File:Joel_spolsky_on_20_sept_2007.jpg)

# Thank you!

@jugmac00 / @jugmac00@fosstodon.org  
<https://jugmac00.github.io>



Canonical