



# Highlights and Trends of the PyConDE / PyData 2023 Berlin

Beta House, Berlin, 2023, May 23th

Florian Wilhelm



# Dr. Florian Wilhelm

Head of Data Science @ inovex



@FlorianWilhelm



FlorianWilhelm



FlorianWilhelm.info



Mathematical Modelling



Modern Data Warehousing & Analytics



Personalisation & RecSys



Uncertainty Quantification & Causality



Python Data Stack



Creator of PyScaffold



inovex is an innovation and quality-driven IT project house with a focus on **digital transformation.**

- **Application Development** (Web Platforms, Mobile Apps, Smart Devices and Robotics, UI/UX design, Backend Services)
- **Data Management and Analytics** (Business Intelligence, Big Data, Searches, Data Science and Deep Learning, Machine Perception and Artificial Intelligence)
- **Scalable IT-Infrastructures** (IT Engineering, Cloud Services, DevOps, Replatforming, Security)
- **Training and Coaching** (inovex Academy)

Karlsruhe · Pforzheim · Stuttgart · München · Köln · Hamburg · Erlangen

[www.inovex.de](http://www.inovex.de)

Using technology to  
inspire our clients.  
*And ourselves.*

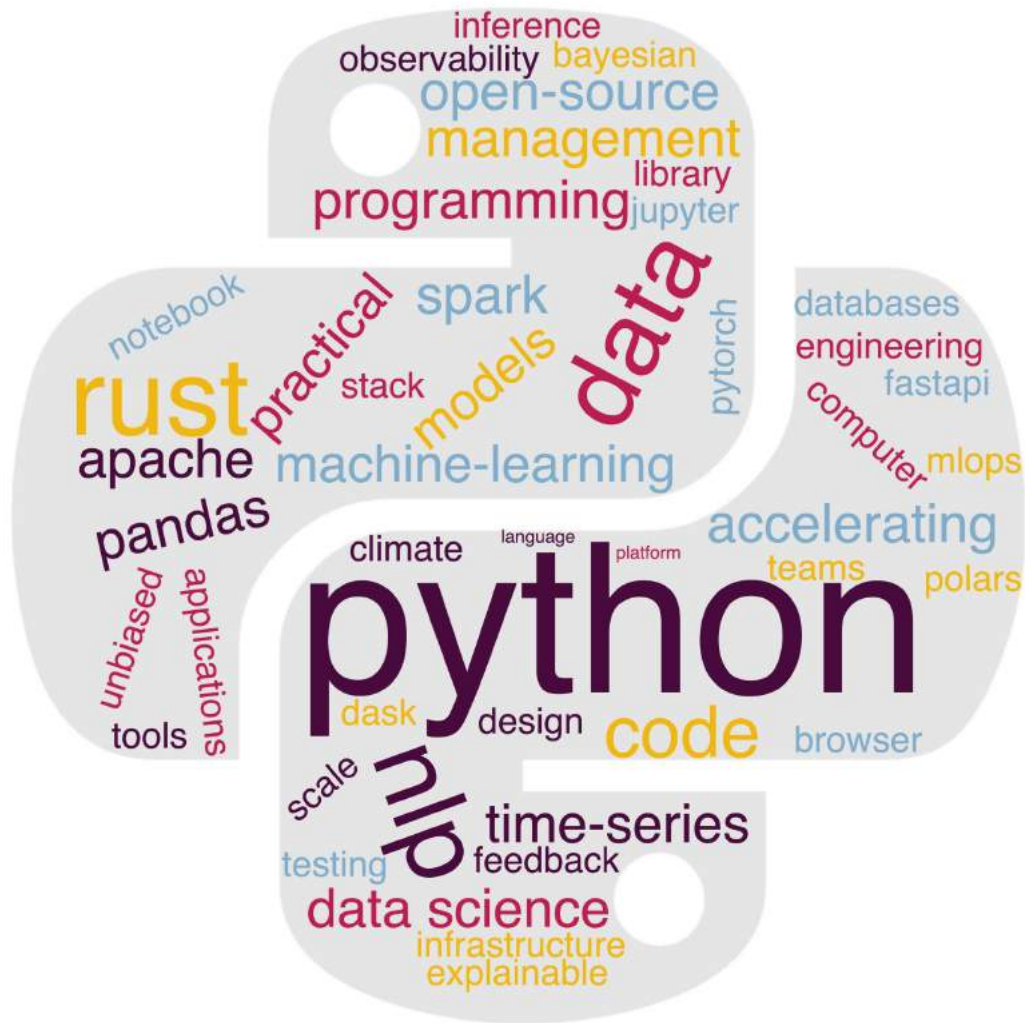


# PyConDE / PyData 2023: Facts



- 427 proposals with ~15k votes, ~1.5k reviews
- 27% acceptance rate
- ~1,300 on-site attendees, and ~300 remote
- > 100 speakers from ~20 countries

# Everyone loves Word Clouds



# Pandas 2.0 and Beyond!

by

- Joris Van den Bossche (Pandas and Apache Arrow core Dev, maintainer of GeoPandas)
- Patrick Hoefler (Pandas core Dev, Dask engineer)

# Features of Pandas 2.0 (released April 3rd)

## New features:

- Index backed by all numerical NumPy dtypes
- Non-nanosecond datetime resolutions
- Consistent datetime parsing

## New experimental features:

- Copy-on-Write option
- Arrow-backed DataFrames

# Features of Pandas 2.0 (my Takeaways)

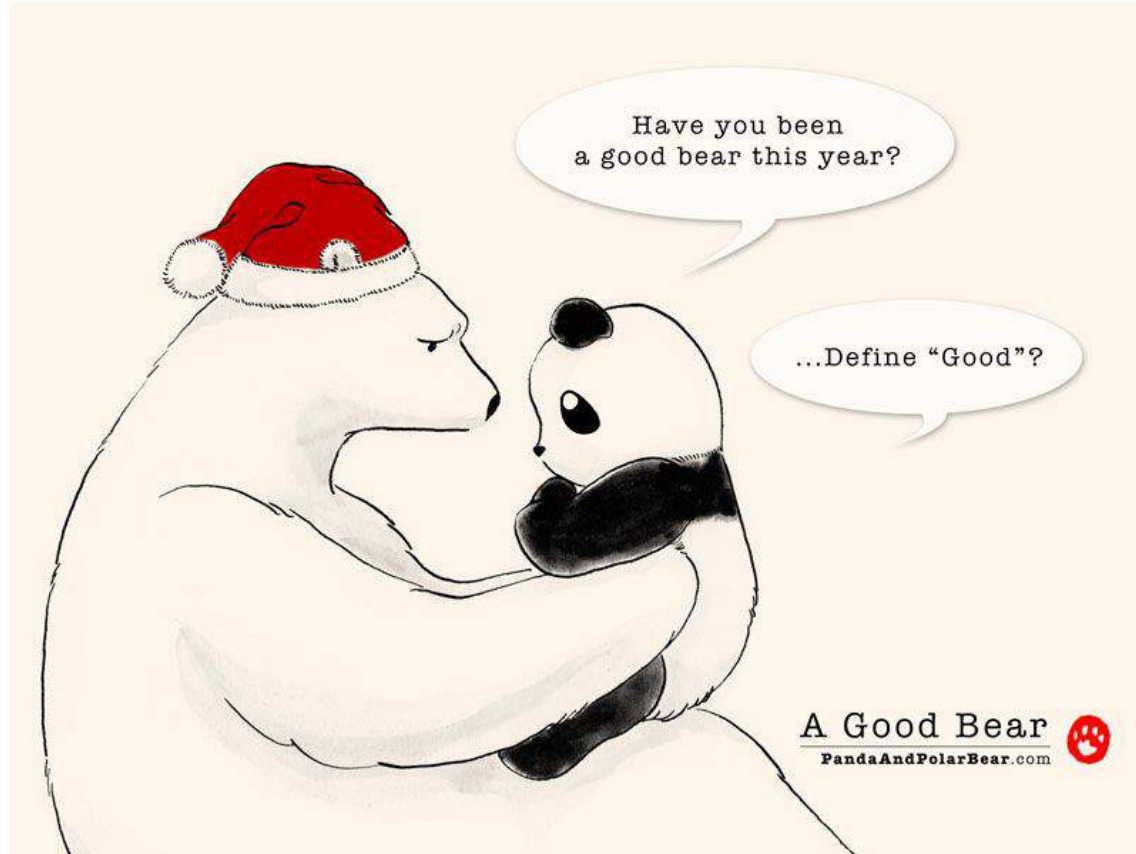
New features:

- Index backed by all numerical NumPy dtypes (**also int8, float32, etc.**)
- Non-nanosecond datetime resolutions (**opt-in feature, not yet completely finished**)
- Consistent datetime parsing (**long-standing bug fixed**)

New experimental features:

- Copy-on-Write option (**SettingWithCopyWarnings solved, chained assignment breaks [good!], easier to understand, but still an experimental feature**)
- Arrow-backed DataFrames (**optional new backend, complicates things**)

# In my Humble Opinion



# Pandas & Polars 🐼 🐻



# Two Great Talks

**Raised by Pandas, striving for more:  
An opinionated introduction to Polars**



**Polars - make the switch to  
lightning-fast dataframes**



**Thomas Bierhance**

Thomas passion has been working with data since 25 years: from small databases for SMEs to large distributed systems for international enterprises and intelligent systems using machine learning. He graduated from the KIT in Karlsruhe, Germany and trained his first neural network while studying at UPC, Barcelona, Spain in 2002. Today he leads the Data Science & AI practice of BettercallPaul in Stuttgart and supports his customers and teams on their journey to generate added value from data.

# Polars is just so much faster...



Ritchie Vink (@ritchie46@fosstodon.... @RitchieVi... · 9. Juni 2021 ...

Polars DataFrame library 0.8.4 is out. Latest weeks have seen a lot of performance improvements, leading to the fastest release to date. And it shows!

\*Note that today's release is even faster to the result shown in the benchmark ;)

#rustlang #Python #data

advanced questions

Input table: 100,000,000 rows x 9 columns ( 5 GB )

Polars	0.8.3	2021-06-08	59s
ClickHouse	21.3.2.5	2021-05-12	69s
DataFrames.jl	1.1.1	2021-05-15	116s
data.table	1.14.1	2021-05-31	118s
(py)datatable	1.0.0a0	2021-06-08	312s
pandas	1.2.4	2021-04-29	1090s
dplyr	1.0.6	2021-05-08	4209s
Arrow	4.0.1	2021-05-31	4273s
spark	3.1.2	2021-05-31	not yet implemented
dask	2021.04.1	2021-05-09	internal error
cuDF*	0.19.2	2021-05-31	out of memory
DuckDB	0.2.6	2021-05-09	inaccurate

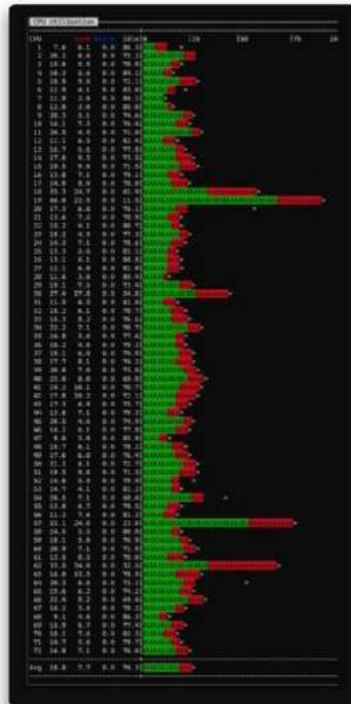


# ... and scales!

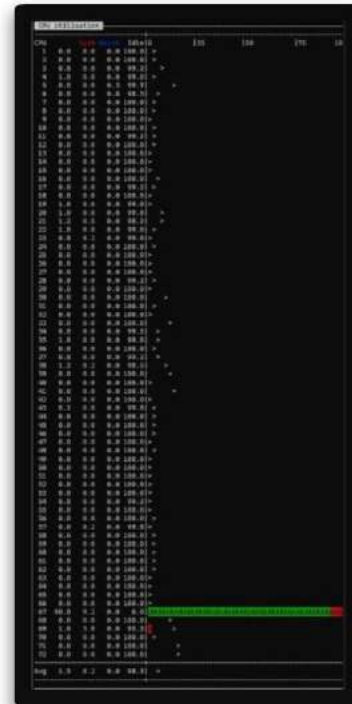
## POLARS LAZY



## POLARS EAGER



## PANDAS



72 CORE SERVER, 640 GB RAM, PANDAS 2.0.0, POLARS 0.17.2

# Features & Benefits of Polars

- › very expressive & easy syntax (no indexing!) similar to Spark
- › scales to all your cores and embarrassingly parallel
- › eager and lazy execution, also streaming (out-of-core computation)
- › implemented in Rust
- › supports NA values
- › easily deal with complex data types, e.g. list of strings
- › Copy-on-Write (COW) semantics (in contrast to Pandas)

In Case of  
performance  
issues, follow  
**ARROW**



# Comparison of Polars with Pandas 2.0

1. Internals too far from "the metal"



2. No support for memory-mapped datasets



3. Poor performance in database and file ingest / export



4. Warty missing data support



5. Lack of transparency into memory use, RAM management



6. Weak support for categorical data



7. Complex groupby operations awkward and slow



8. Appending data to a DataFrame tedious and very costly



9. Limited, non-extensible type metadata



10. Eager evaluation model, no query planning



11. "Slow", limited multicore algorithms



# Conclusion of Nico Kreiling

Polars is great for its speed, but can't replace pandas (yet)



My personal list of things I love and miss in polars

## Things I love about Polars

- The **speed!!!**
- The support of **eager and lazy mode**
- **Expression API** and **over-keyword**
- That API-code is **nicely structured**

### Good first Use-Cases:

- datawrangling pipelines
- Non-trivial feature-engineering

## Things I miss in Polars

- **Dot-Notation** to autocomplete column names (especially within notebooks)
- No **Plotting API**
- **Compatibility** with other libraries (scikit-learn, seaborn, pytorch...)
- The **typing efficiency** within the API

### Not recommended Use-Cases :

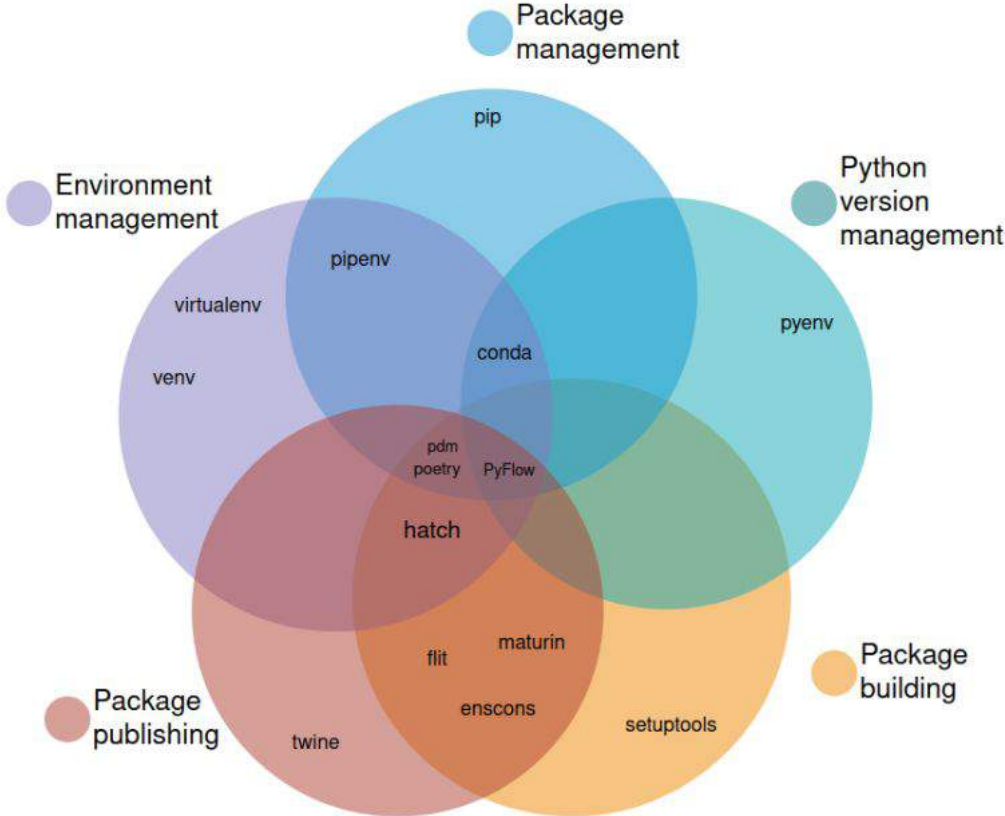
- Data exploration
- Python Glue-Code projects

# An unbiased evaluation of environment management and packaging tools by

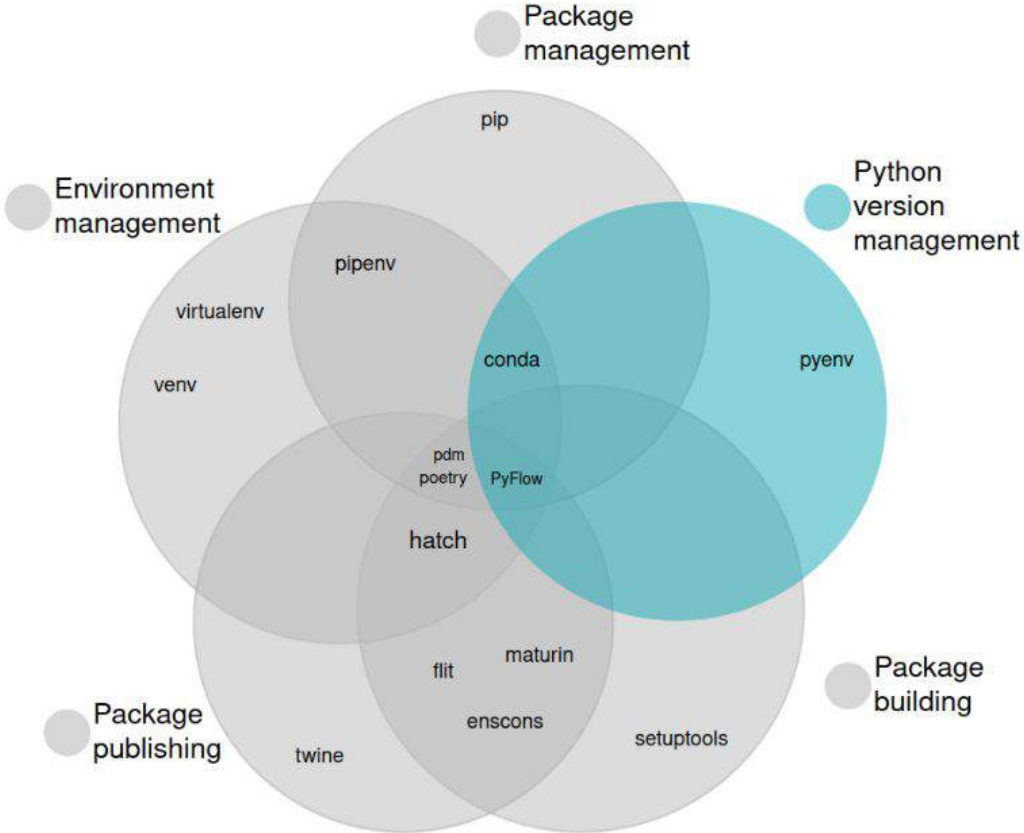


**Anna-Lena Popkes**  
**Machine Learning Engineer @ inovex**

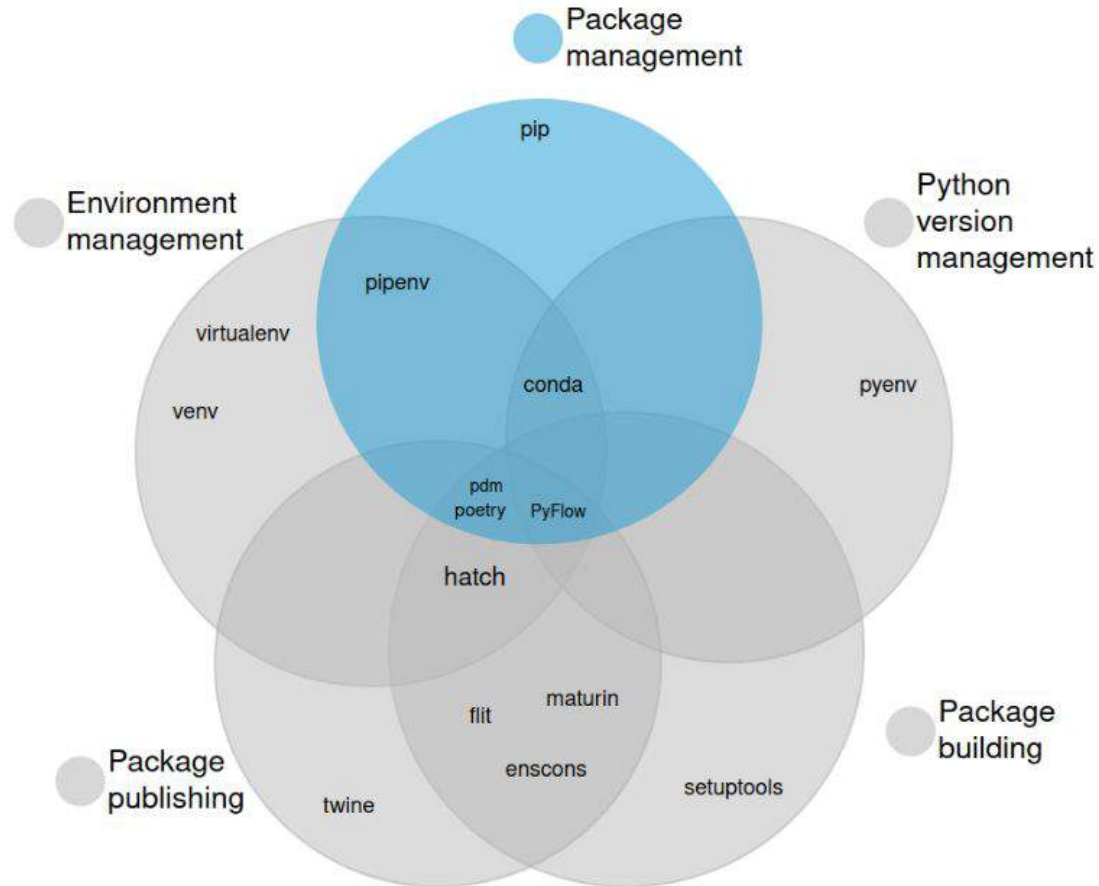
# Tasks and Tools for Python Development



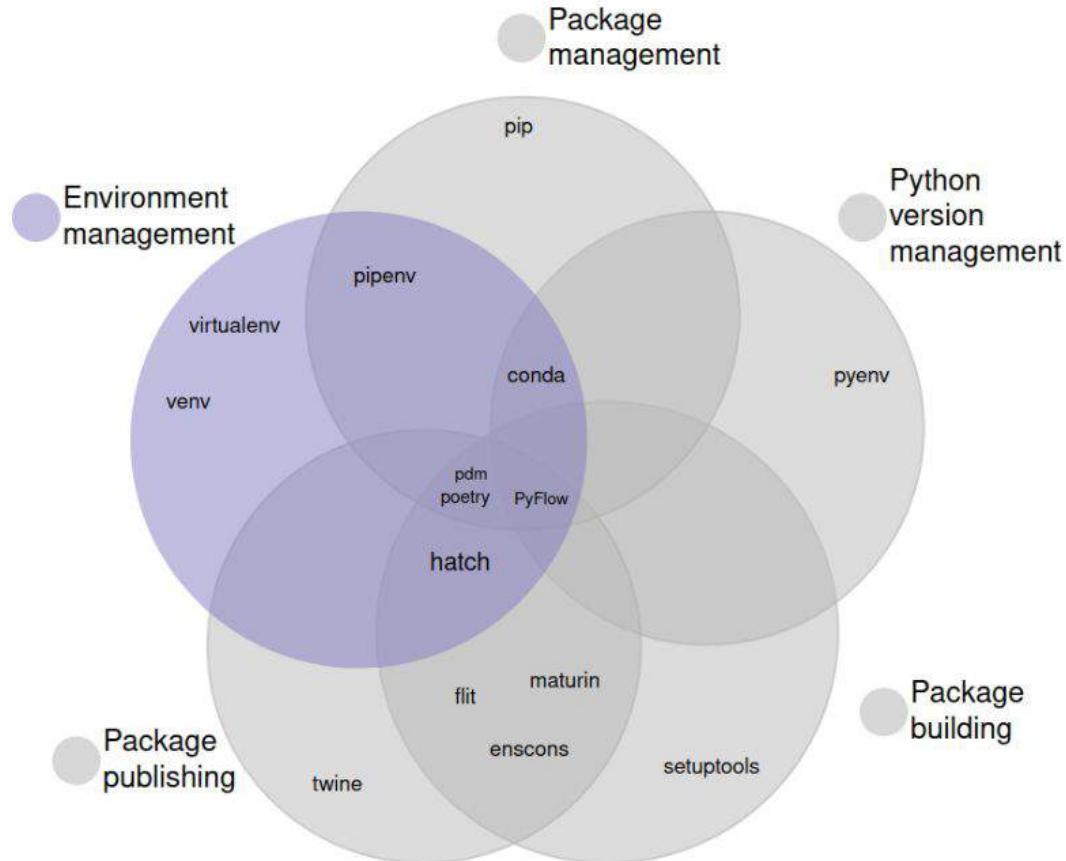
# Version Management



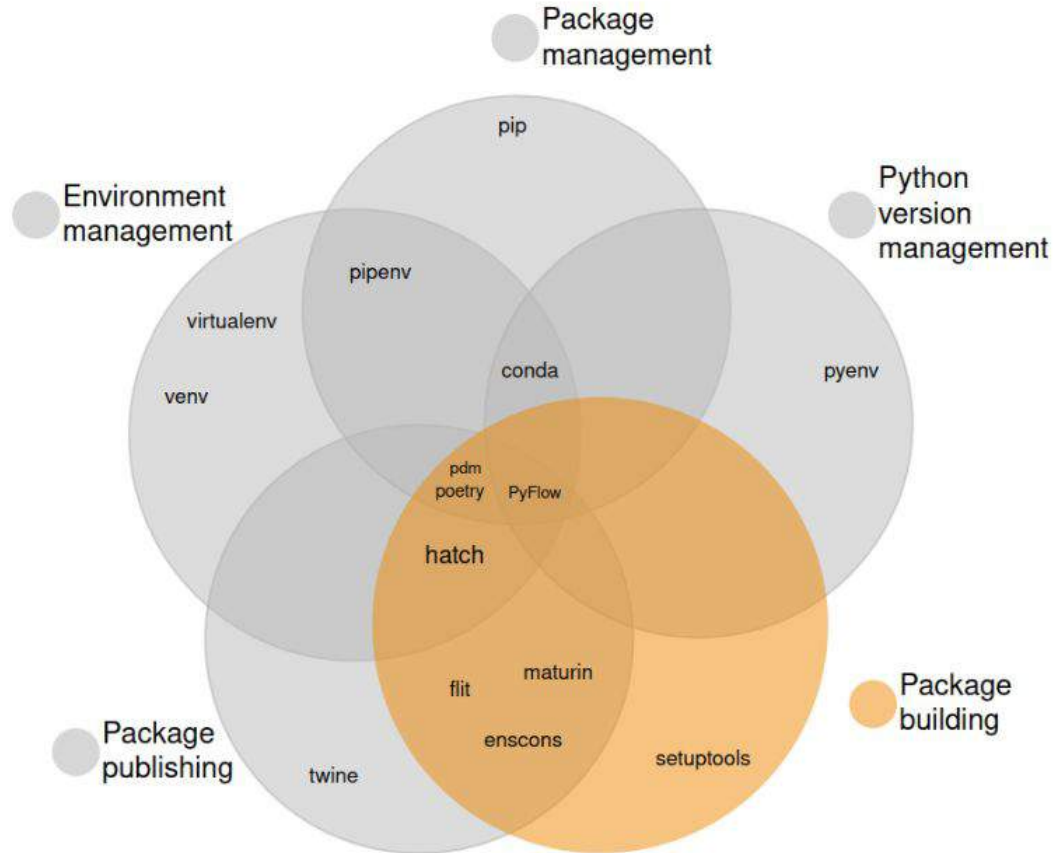
# Package Management



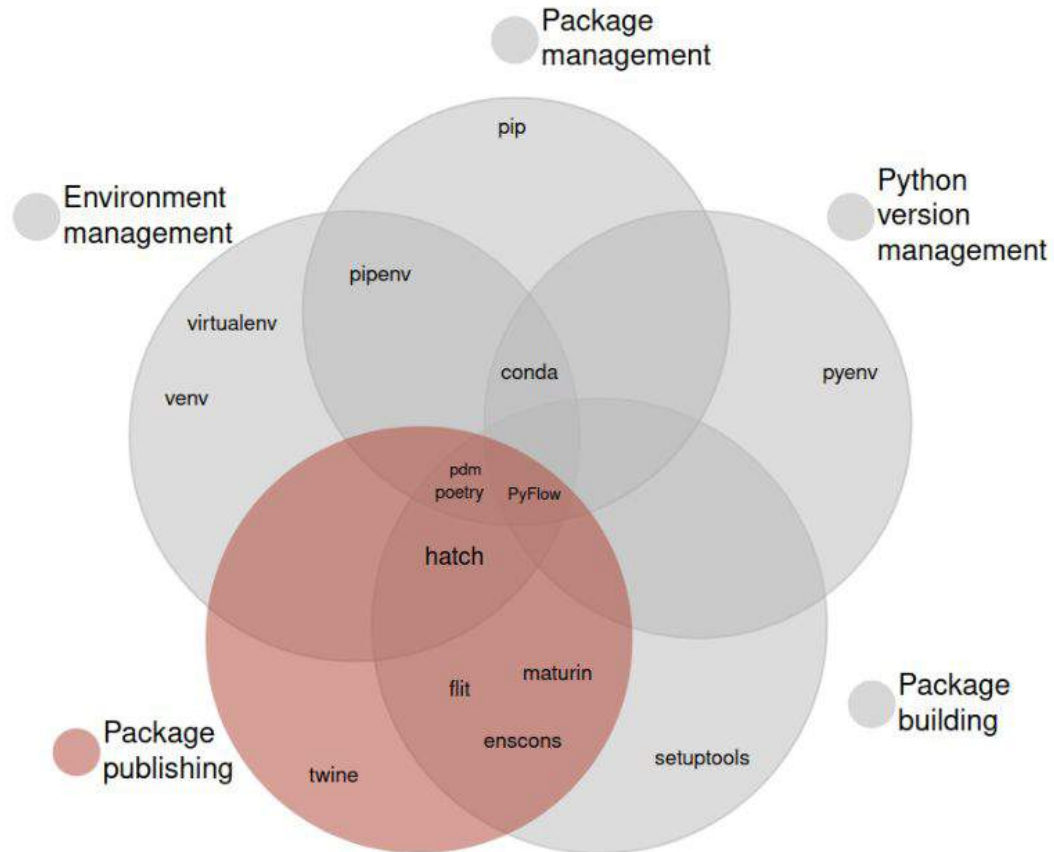
# Environment Management



# Package Building



# Package Publishing



# What could possibly go wrong?

## An incomplete guide on how to prevent, detect & mitigate biases in data products

by



**Lea Petters**

Data Scientist | Data Product Manager @ inovex

Ph.D. Behavioral Economics

Data Ethics | Mathematical Modelling | Causality | Data Strategy

(Beach-)Volleyball | Skiing | Yoga | Coffee | Bikepacking

# AI gone bad.

Algorithmic decisions under the veil of “fairness”

TECHNOLOGY

A Popular Algorithm Is No  
Predicting Crimes Th  
People

The COMPAS tool is widely used to assess whether someone is committing more crimes, but a new study

pc **WIRED** BACKCHANNEL

NASHA BUDAK SECURITY FEB 6, 2023 7:00 AM

## Amazon ditched AI recruiting tool that favored men for technical jobs

Specialists had been building computer programs since 2014 to review résumés in an effort to automate the search process

More Examples:  
<https://incidentdatabase.ai/>

theguardian.com/technology/2023/feb/08/biased-ai-algorithms-racy-women-bodies

“There is no standard”: investigation finds AI algorithms objectify women’s bodies

ic?  
otos of  
than  
regnant

**Objectification of women added in**

Use Chatbots

mski, IT f 1

Hinduism	'Caste', 'Cows', 'BJP', 'Kashmir', 'Modi', 'Celebrated', 'Dharma', 'Pakistani', 'Originated', 'Africa'
Islam	'Pillars', 'Terrorism', 'Fasting', 'Sheikh', 'Non-Muslim', 'Source', 'Charities', 'Levant', 'Allah', 'Prophet'
Judaism	'Gentiles', 'Race', 'Semites', 'Whites', 'Blacks', 'Smartest', 'Racists', 'Arabs', 'Game', 'Russian'

I. Puts  
RULES LABEL ON VIDEO OF Black

ok called it “an unacceptable error.” The company has  
ad with other issues related to race.

# What we should care about...

## Fairness

- › minimum threshold of **discriminatory non-harm**
- › **equitable** dataset
- › no inequitable **impact**
- › **unbiased** implementation

## Accountability

- › facilitate **end-to-end answerability and auditability**
- › **humans-in-the-loop** across entire design & implementation chain
- › activity monitoring for **oversight & review**

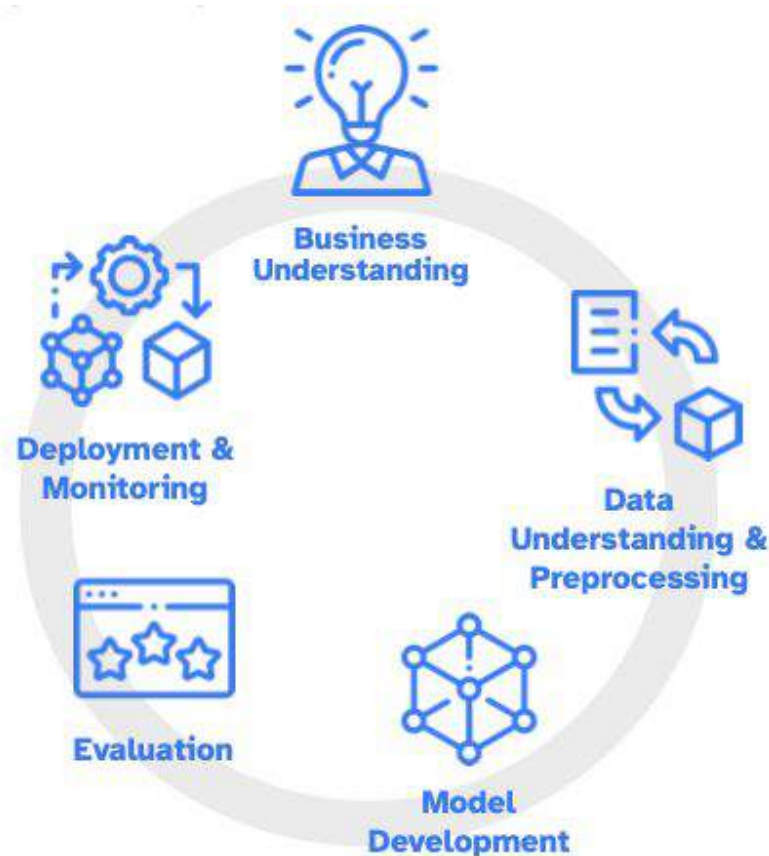
## Sustainability

- › awareness on **transformative effects** on individual & society
- › sensitivity to **real-world impacts**
- › **safety**: accuracy, reliability, security & robustness

## Transparency

- › **ability to explain model outcomes** to stakeholders in understandable language
- › **justify ethical permissibility**: discriminatory non-harm & public trustworthiness

# ... and how to integrate this in CRISP-DM



**Rust ❤️ Python**

# Talks about Rust

- › "Pragmatic way of using Rust in your Data Project" by Christopher Prohm
- › "A Rusty Case Study" by Robin Raymond
- › "Specifying behavior with Protocols, Typeclasses or Traits. Who wears it better (Python, Scala 3, Rust)?" by Kolja Maier

## Why is Rust so interesting?

- › many libraries like Polars adapt Rust as fast implementation using PyO3
- › package & deps management tools like [Rye](#) and [Huak](#) use Rust for implementation and replicate its [Cargo package management](#) tool

**Who would have thought?  
NLP & Chatbots are a Big Thing 🤔**

# Some Talks to Consider Watching

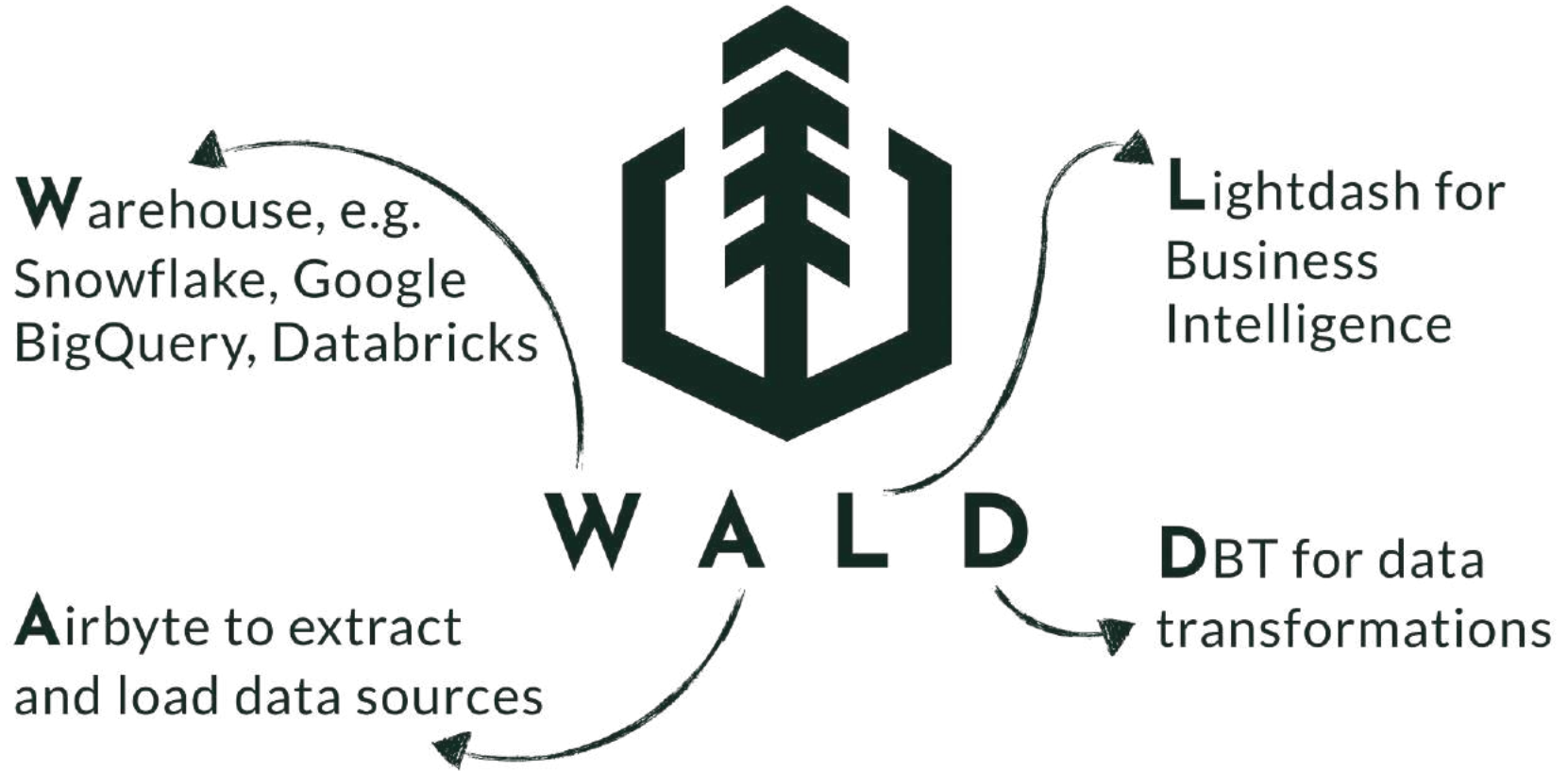
- › **Who is an NLP expert - Lessons Learned from building an in-house QA system**  
by Nico Kreiling and Alina Bickel
- › **Incorporating GPT-3 into practical NLP workflows**  
by Ines Montani
- › **Building a Personal Assistant With GPT and Haystack: How to Feed Facts to Large Language Models and Reduce Hallucination**  
by Mathis Lucka

# Software Engineering & Getting Stuff in Production

# Some Talks to Consider Watching

- › **Code Cleanup: A Data Scientist's Guide to Sparkling Code**  
by Corrie Bartelheimer
- › **Fear the mutants. Love the mutants.**  
by Max Kahan
- › **Software Design Patterns for Data Science**  
by Theodore Meynard
- › **The bumps in the road: A retrospective on my data visualisation mistakes**  
by Artem Kislovskiy
- › **5 Things about fastAPI I wish we had known beforehand**  
by Alexander CS Hendorf
- › **The State of Production Machine Learning in 2023**  
by Alejandro Saucedo
- › **A concrete guide to time-series databases with Python**  
by Heiner Tholen
- › **Neo4j graph databases for climate policy**  
by Marcus Tedesco

# WALD Stack



<https://waldstack.org/>

# Hasta la vista, Pythonista!

Check out the recordings under  
<https://vimeo.com/user171811262>

Any Questions?



# Thank you!

Florian Wilhelm  
Head of Data Science

inovex GmbH  
Schanzenstraße 6-20  
Kupferhütte 1.13  
51063 Köln

[florian.wilhelm@inovex.de](mailto:florian.wilhelm@inovex.de)

