

# Reproducibility Engineering Portfolio Exam

artificial business scenario „ducks vs. fish“

Sebastian Sippl  
Thomas Brandl

# Table of contend

- Variables & Optimization
- Feasible Region & Excel Solver
- Reproduction Package
- Decoupling Dependency
- Docker
- Zenodo & GitHub
- Live demonstration

Optimizing the product mix for rubber ducks and fish. (“Head First Data Analysis”)

→ Maximize the profit

Constrains:

- Variables which can’t be controlled and therefore limit the output

Decision Variables:

- Variables which can be controlled and actively changed

Optimization Problem:

In general:  $P = c_1 * x_1 + c_2 * x_2$

In our scenario:  $P = p_d * x_d + p_f * x_f$

with  $p := Profit$ ,  $x := Number$   
 $f := Fish$ ,  $d := Ducks$

## Original experimental setup

Rubber subbly for max.

500 ducks & 0 fishes or  
400 fishes & 0 ducks

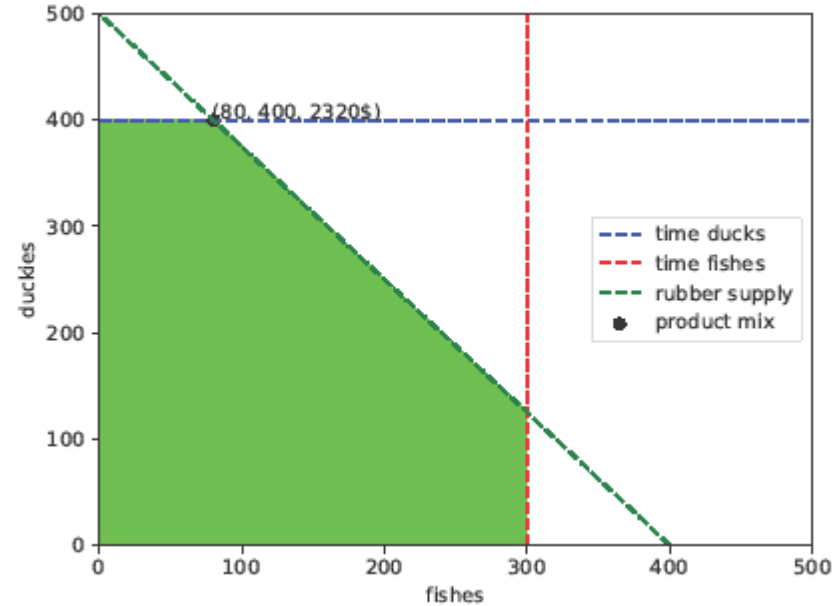
Production Time for max.

400 ducks and  
300 fishes

Profits:

one Duck: 5\$

one Fish: 4\$

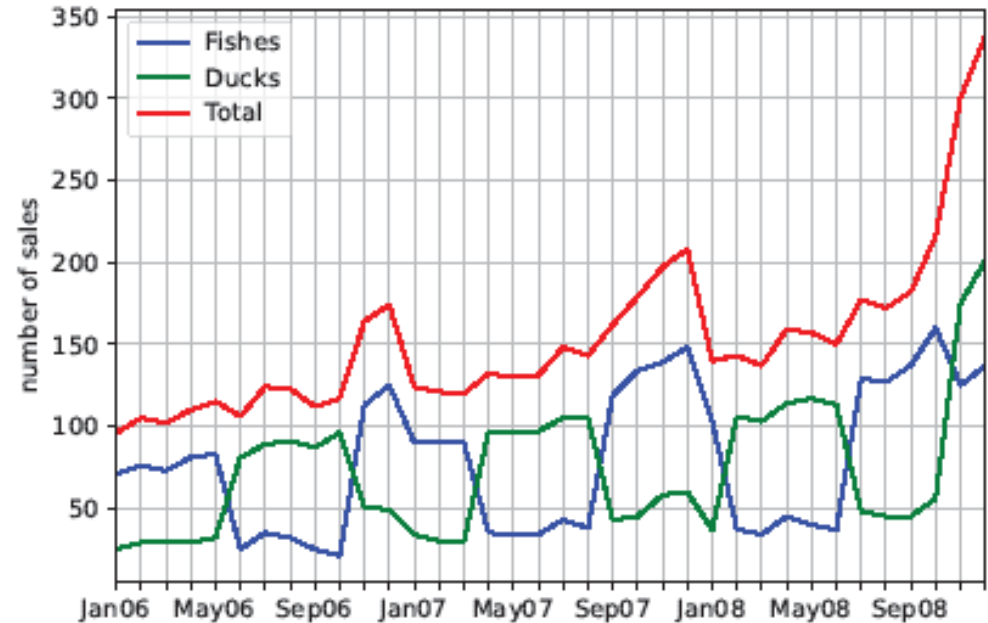


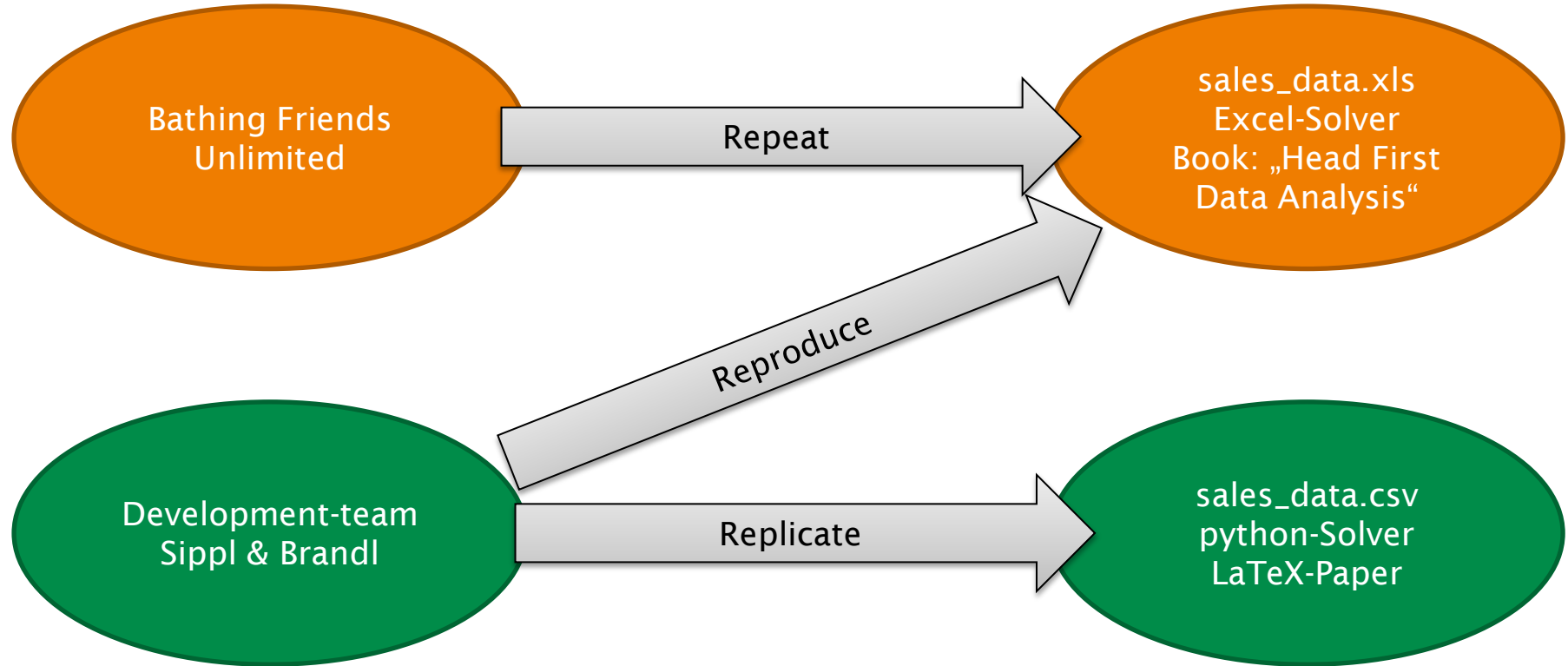
Best Product-Mix:

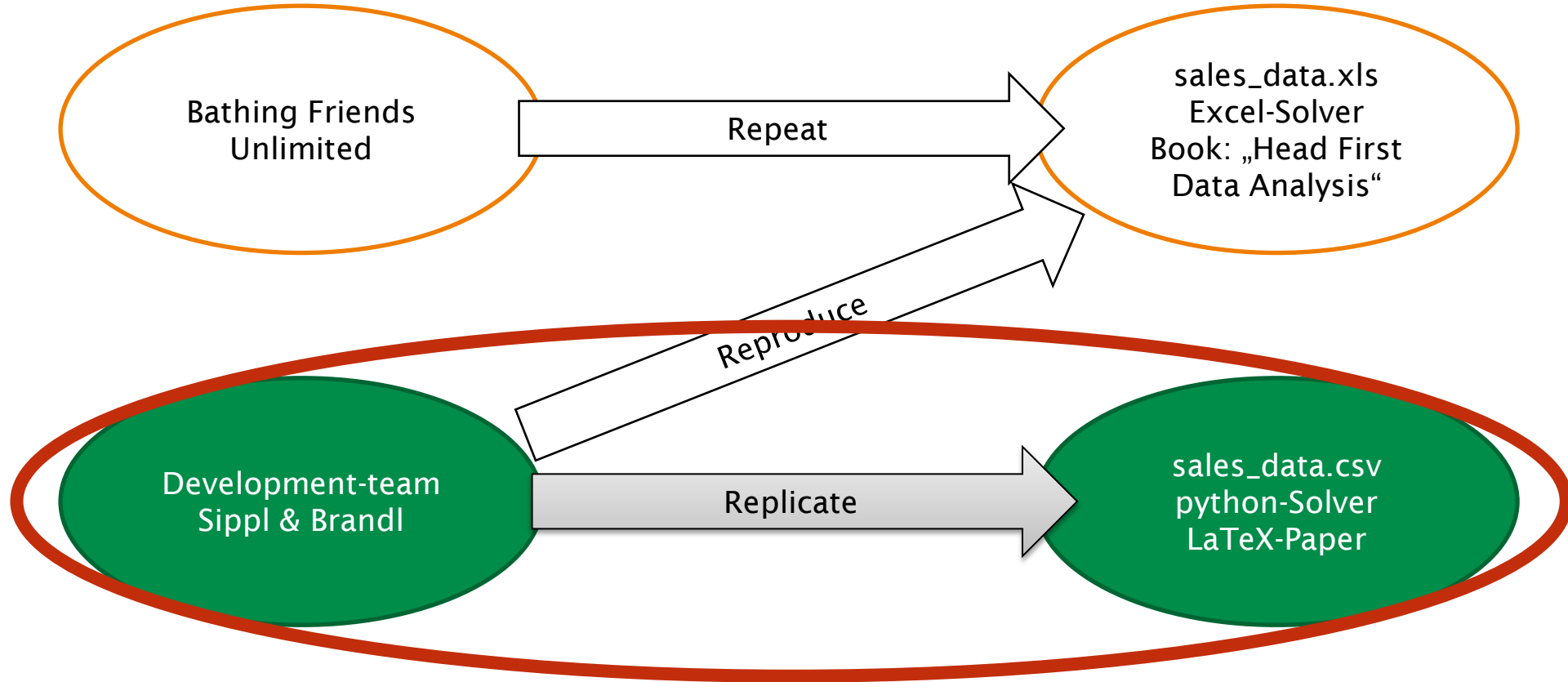
$$80 \text{ fishes} * 4\$ + 400 \text{ ducks} * 5\$ = 2320\$$$

## Historical sales data

Historical sales data about 36 Month  
Data format: EXCEL-file (.xls)







## Avoid proprietary software

The main advantages of open source software:

- Long term availability
- No missing licenses or avoid fee from licensing the tools
- Open source alternative for most common tools
  - MS-Word -> LaTeX
  - MS-Excel -> R, Python.Numpy/.Matplotlib



## Tools in our experiment

Realize the Excel Solver via python

- computes every possible product mix
- Beneficial for researchers who repeat our experiment and have no access to MS-Tools.

Exchange xls files with csv files

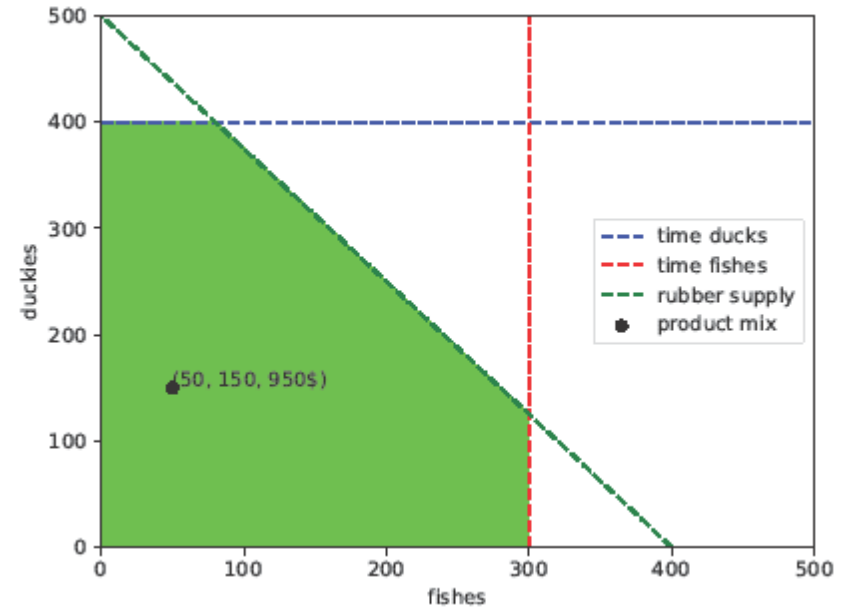
- light-weighted data format
- easy to read for humans (plain text)
- csv is supported by many scientific applications

Realize the documentation via LaTeX

- Documentation is generate automatically
- best look for scientific papers
- automatically import of figures

## Reproducibility results

- Solved with python-script
- Able to edit the constraints via comand line interface
- Same constraints like the origin experiment + prediction constraint from historical data



## Build and run the experiment

- Docker is used as encapsulated environment (container)
  - For developing, shipping and running applications
  - Easy to share the same content
  - An image is a read only template
  - A container is a runnable instance of the image
- The container of the repro-package:
  - Holds the whole setup (data, sourcecode, doku)
  - During build-process all tools are installed and the setup is copied into
  - The experiment starts automatically
  - The documentation are generated after that

## History and long term availability

Git and GitHub:

- For a traceable project history
- To organize the project in the team
- No guarantee for long-term documentation at GitHub

Zenodo

- financed by public funds from the EU
- Link to a releases of the project/experiment
- referenceable by other researchers with a DOI

# Live demonstration of the experiment

# References

- [1] Michael Milton, "Optimization: Take It to the Max," in Head First Data Analysis, O'Reilly Media Inc, 2009, pp. 75-109
  
- [2] Microsoft, 2022, <https://support.microsoft.com/enus/office/define-and-solve-a-problem-by-usingsolver-5d1a388f-079d-43ac-a7eb-f63e45925040> (accessed on 03.02.2022)
  
- [3] Docker Docs, 2021, <https://docs.docker.com/getstarted/overview/> (accessed on 13.02.2022)
  
- [4] The Matplotlib Development team, 2021 <https://matplotlib.org/> (accessed on 17.02.2022)
  
- [5] GitHub, Inc., 2022, <https://github.com/> (accessed on 17.02.2022)
  
- [6] CERN Data Centre Invenio., 2022, <https://zenodo.org/> (accessed on 17.02.2022)