

Jan Dietrich

# Open Source Software Github Benchmark

OSS Benchmark

[Github Source Code](#) 

[Visualisierung](#)

# OSS Benchmark Ziele

Informationen zu den Schweizer Open Source Beiträgen auf Github

- Datensammlung und Aufbereitung
- Visualisierung der Daten
- Download der Daten

# Prinzipien

- Daten
  - vollständig und korrekt
  - sauber strukturiert
  - als CSV und JSON verfügbar
- Visualisierung mit D3
  - möglichst einfach
  - erweiterbar
  - ~~keine Verwendung von weiteren Frameworks~~

# Vorgehensweise

1. **Github Organisationen** - Spezifizieren welche Organisationen auf Github aus der Schweiz sind
2. **Github API** - Datensammlung mit Hilfe von Python
3. **Visualisierung erstellen** - Visualisierung mit D3
4. **Visualisierung bereitstellen** - Github Actions

# Schweizer Organisationen Github

- Manueller Prozess
- Einpflegen in JSON - [/data-gathering/github\\_repos.json](/data-gathering/github_repos.json)

```
{ "GitHubRepos":  
  {  
    "IT": {  
      "name": "Informatikbranche",  
      "institutions": [  
        { "name": "4teamwork AG", "orgs": ["4teamwork"] },  
        ...  
      ],  
      "ResearchAndEducation": {  
        "name": "Hochschulen und Forschung",  
        "institutions": [  
          { "name": "Universität Bern", "orgs": ["digital-sustainability"] },  
          ...  
        ]  
      }  
    }  
  }  
}
```

# Daten von Github

## Github API [Link](#)

```
GET https://api.github.com/orgs/digital-sustainability/repos
```

```
[{  
  "name": "oss_directory",  
  "language": "JavaScript",  
}]
```

## PyGithub

```
g.get_organization('digital-sustainability').get_repo('oss_directory').language
```

```
'JavaScript'
```

# Daten von Github

## Organisation, welche Kennzahlen?

- Anzahl Repositories
- Commits
- Stars
- Pull Requests
- Issues ...

→ [Jupyter Notebook](#) 

# Daten von Github

## Skript

[/data-gathering/OSS\\_github\\_benchmark.py](#)

```
for organisation in organisationen:
    ...
    for repository in organisation.repositories:
        datensammlung['stars'] = repository.stargazers_count
        datensammlung['watchers'] = repository.subscribers_count
    ...

datensammlung.toJSON()
datensammlung.toCSV()
```



# Daten von Github

## Problem

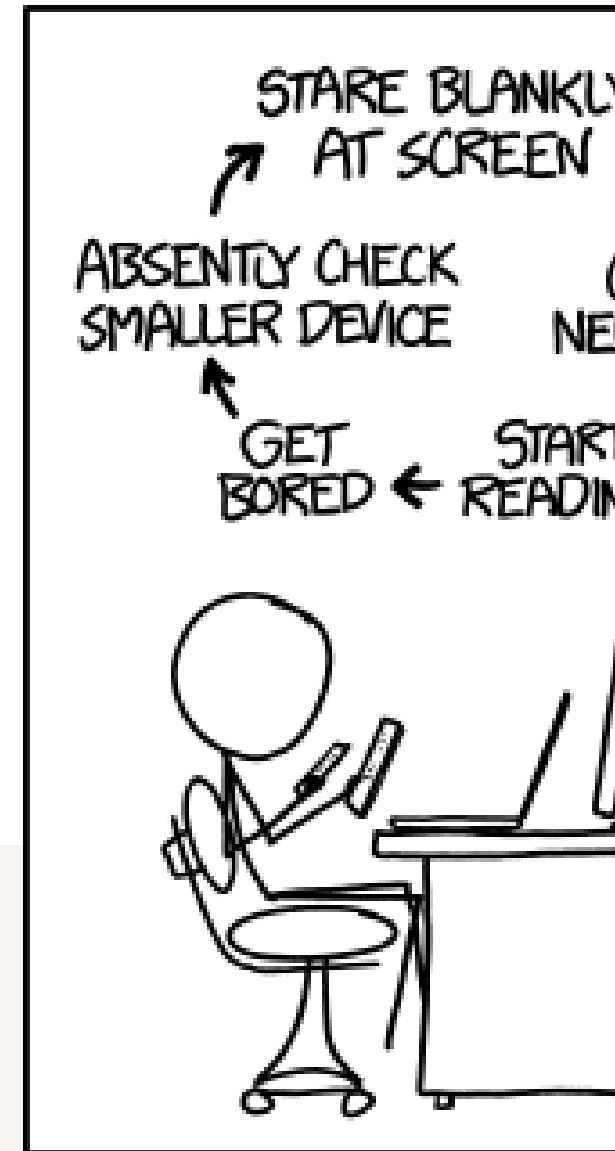
### Github Rate Limit

Nur 5'000 Requests pro Stunde

## Lösung

Immer wieder eine Stunde warten

```
def handle_rate_limit():  
    reset_time = datetime.datetime.fromtimestamp(g.rate_limiting_resettime)  
    logger.warning(f'rate limit exceeded, continuing on {reset_time}')  
    while datetime.datetime.now() < reset_time:  
        sleep(1)
```



# Visualisierung

[OSS Benchmark](#)

# Visualisierung bereitstellen

## Github Actions

digital-sustainability / oss-github-benchmark

<> Code

! Issues 4

🔗 Pull requests 4

▶ Actions

📁 Projects

📖 Wiki

🛡 Security

📊 Insights

⚙ Settings

Unwatch 4

Workflows

New workflow

All workflows

🔗 Test input data

🔗 Build and Deploy App

🔗 Python package

🔗 Publish Docker image

All workflows

🔍 Filter workflows

96 results

Event ▾

Status ▾

Branch ▾

Actor ▾

● Python package

Python package #61: Scheduled

📅 12 hours ago

🕒 Queued

...

✖ Python package

Python package #60: Scheduled

📅 2 days ago

🕒 6h 0m 25s

...

✖ Python package

Python package #59: Scheduled

📅 3 days ago

🕒 6h 0m 28s

...

✖ Python package

Python package #58: Scheduled

📅 4 days ago

🕒 6h 0m 26s

...

✖ Python package

Python package #57: Scheduled

📅 5 days ago

🕒 6h 0m 26s

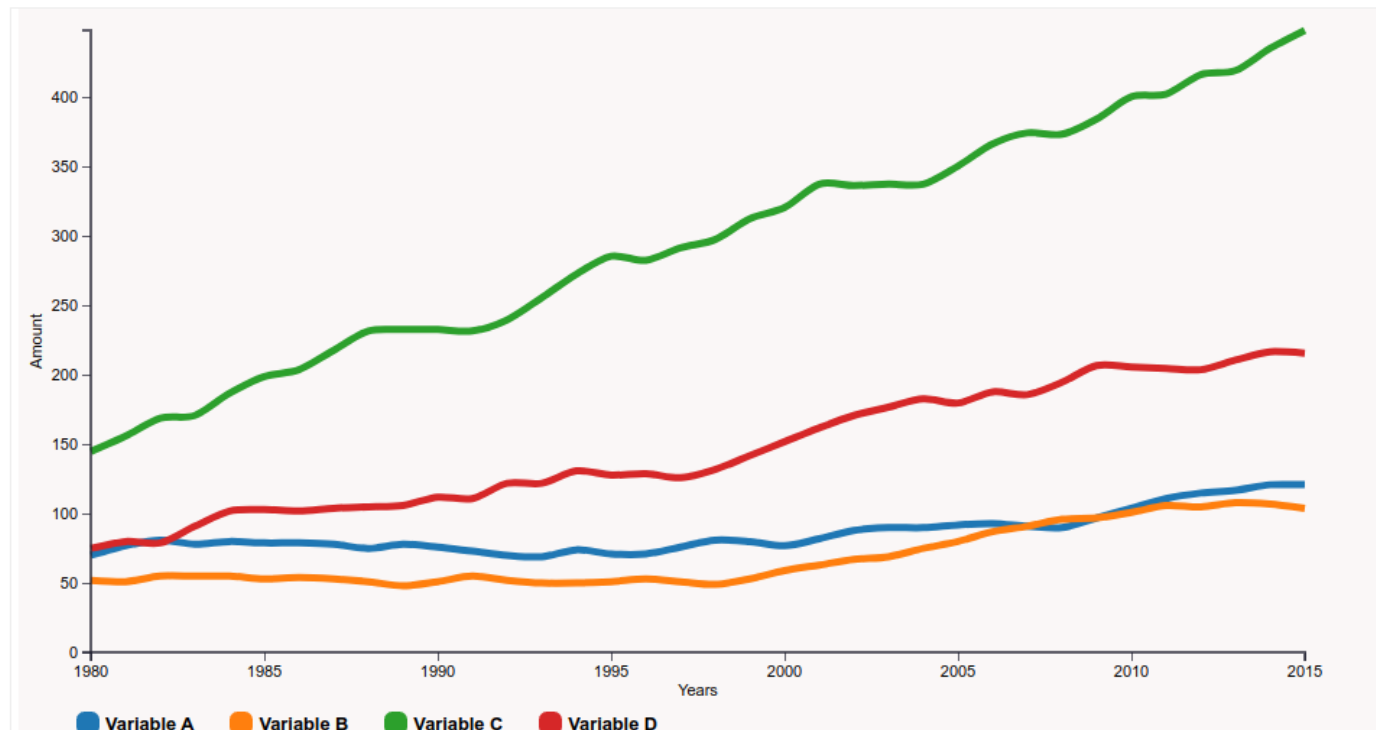
...

# Summary

- Datensammlung
  - PyGithub, Jupyter Notebook, Automatisiertes Script
- Visualisierung
  - D3.js, Angular für den Rest
- Bereitstellung
  - GitHub Actions, GitHub Pages

# Ausblick

- Fix von automatischem Skript
- Tabelle mit Top 20 für eine Kennzahl
- Line Chart mit Commits



# Fragen?


[@CodeOneTwo](#) 

[@CodeOneTwo](#) 

[jan.dietrich@tecpose.com](mailto:jan.dietrich@tecpose.com)

[CodeOneTwo](#)

# Tools und mehr

- `mdx-deck` - for the presentation ([GitHub](#)  )
- `jupyter notebook` - for the documentation ([Jupyter Notebook](#))
- `PyGithub` - library to query GitHub API ([Jupyter Notebook](#))

→ Alles auf Github

- [Link](#) zu der Präsentation
- [Link](#) zu den Jupyter Notebooks