Interactive species habitat analysis

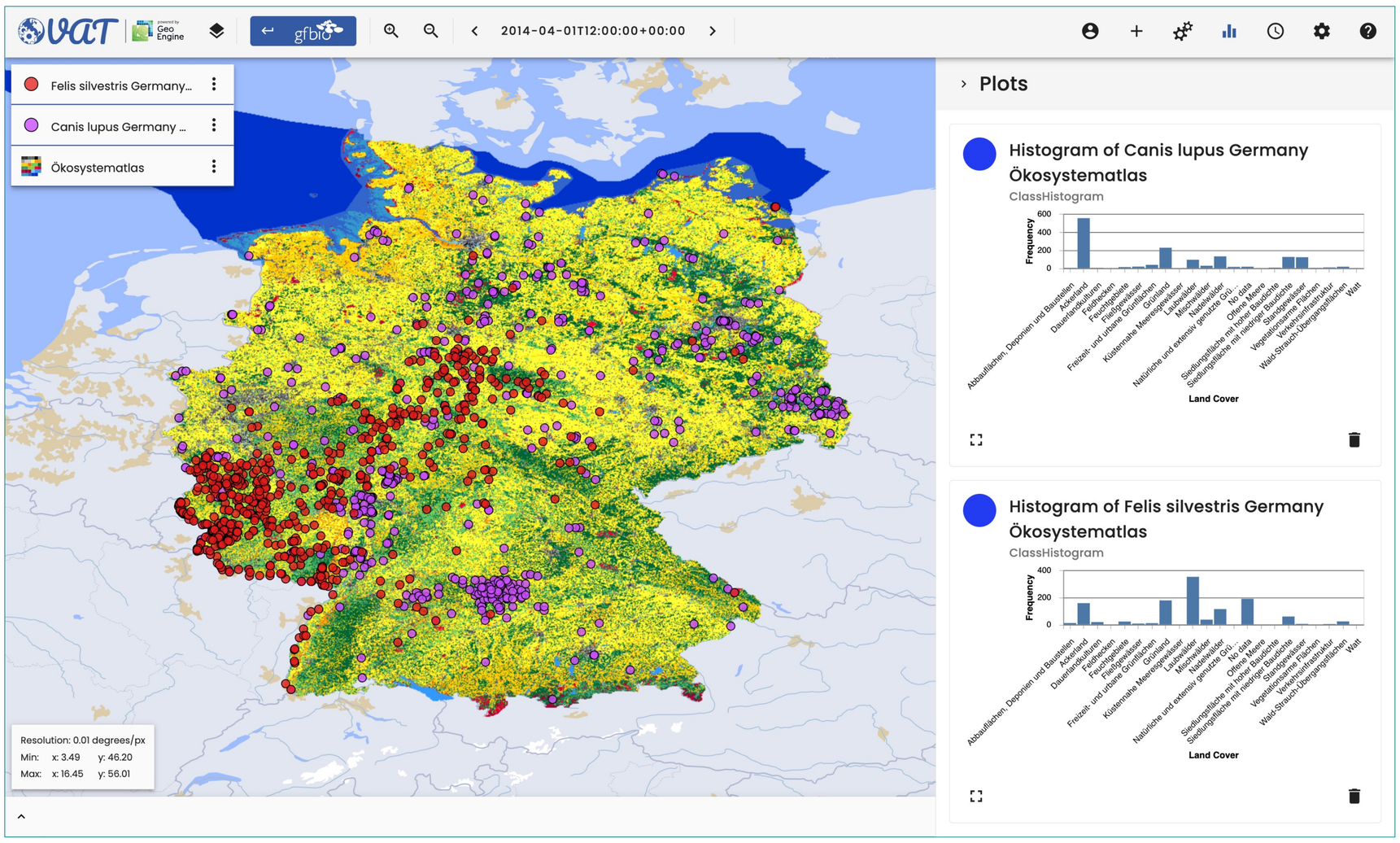
Dominik Brandenstein, Henri Dümpelmann, Bernhard Seeger

Philipps Universität Marburg

Overview

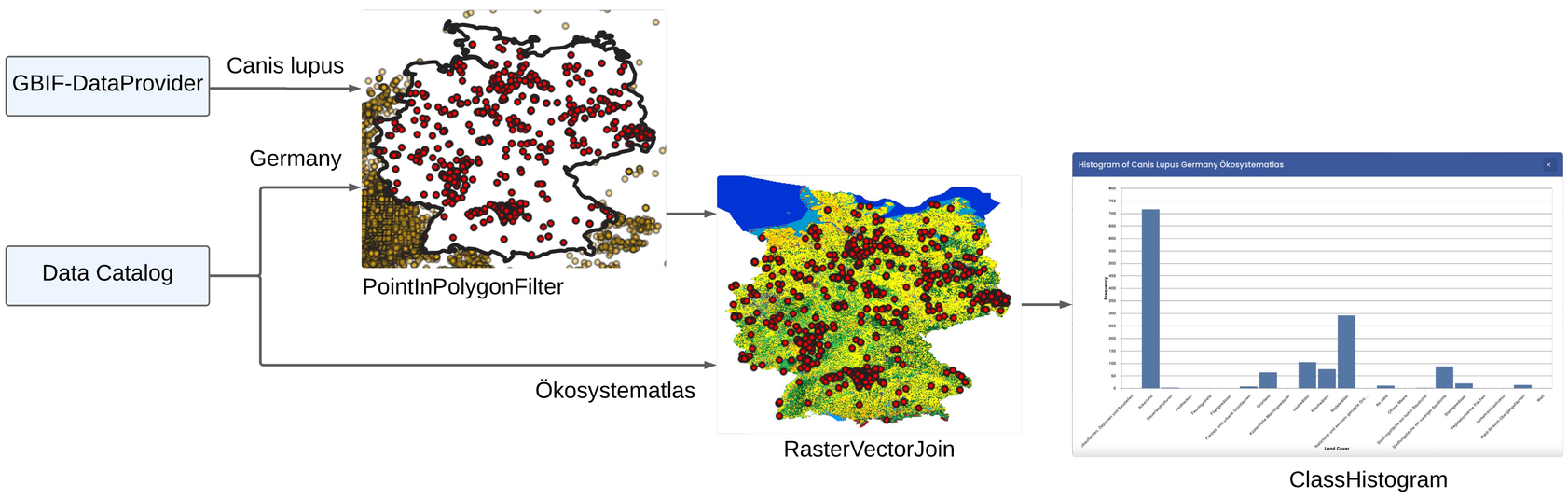
VAT is a web-based GIS application backed by Geo Engine that enables researchers to visualize, analyze, and transform geospatial vector and raster data. The application allows for the creation of complex workflows with ease. An example use-case, 'Canis Lupus meets Felis silvestris,' demonstrates the ability to quickly and easily create an interactive habitat analysis based on the Ökosystematlas in just a few steps.

VAT | powered by Geo Engine



Use-Case Workflow

VAT provides integrated data sources and workflows with available operators for processing and directing the data flow. This approach enables dynamic and interactive calculations. In this example, we use the occurrences of Canis lupus and Felis silvestris from the GBIF-DataProvider, the boundaries of Germany and the Ökosystematlas land use classification. First, we apply a Point-In-Polygon Filter to select all occurrence records within Germany. Then, the Raster-Vector-Join operator spatially joins the filtered occurrence data with the land classification of the Ökosystematlas. The Class-Histogram operator plots the number of occurrence records for each land use class.



Upcoming Developments

- Example Use-Cases with detailed documentation
- Build-up of the semantic storage
- Multi-Band-Raster Operators
- Data-Management Dashboard
- Data-Streams for machine learning applications

Jupyter Notebook Integration

Besides the VAT visual interface of a Web GIS, the Geo Engine Python package also supports workflows in Python programs, e.g., in Jupyter Notebooks. Users can seamlessly exchange their workflows between the Web GIS and the Python package via their unique identifiers. In both tools, users are allowed to extend and modify their workflows.

Imports

```
[1]: #pip install git+https://github.com/geo-engine/geoengine-python
```

```
[2]: import geoengine as ge
from datetime import datetime
import matplotlib.pyplot as plt
import altair as alt
```

Initialize

```
[3]: ge.initialize("https://vat.gfbio.org/api", token="c8bc251d-00cf-4400-bf87-8dce8a0e9313")
```

Get Workflows

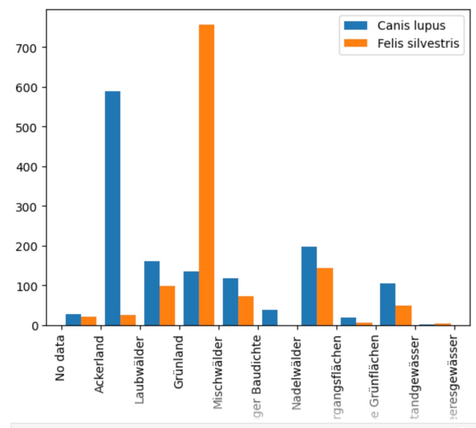
```
[4]: canis_lupus = ge.workflow_by_id("7857aefa-4118-5f6a-8823-51c1aa2aa18e")
felis_silvestris = ge.workflow_by_id("e09e32f8-3704-57a7-903b-10b36a4cbd8e")
```

Query data

```
[5]: time = datetime.strptime(
    '2024-04-25T12:00:00.000Z', "%Y-%m-%dT%H:%M:%S.%f%z")

canis_lupus_data = canis_lupus.get_dataframe(
    ge.QueryRectangle(
        ge.BoundingBox2D(5.852490, 47.271121, 15.022059, 55.065334),
        ge.TimeInterval(time, time),
        resolution=ge.SpatialResolution(1., 1.),
    ),
    resolve_classifications = True
)

felis_silvestris_data = felis_silvestris.get_dataframe(
    ge.QueryRectangle(
        ge.BoundingBox2D(5.852490, 47.271121, 15.022059, 55.065334),
        ge.TimeInterval(time, time),
        resolution=ge.SpatialResolution(1., 1.),
    ),
    resolve_classifications = True
)
```

Plot data

```
[6]: canis_lupus_data = canis_lupus_data['Ökosystematlas']
felis_silvestris_data = felis_silvestris_data['Ökosystematlas']

plt.hist(
    [canis_lupus_data, felis_silvestris_data],
    label=['Canis lupus', 'Felis silvestris']
)
plt.xticks(rotation=90)
plt.legend(loc='upper right')
plt.show()
```




```
[7]: canis_lupus.get_provenance()[:3]
```

```
[7]: [ProvenanceEntry(data=[1c01dbb9-e3ab-f9a2-06f5-228ba4b6bf7a:species/Canis lupus], provenance=Provenance(citation='UMMZ Mammals Data Group, LSA IT A (2023). University of Michigan Museum of Zoology, Division of Mammals. Version 8.55. University of Michigan Museum of Zoology. Occurrence dataset https://doi.org/10.15468/dx3rcj accessed via GBIF.org on 2023-02-17.', license='http://creativecommons.org/licenses/by-nc/4.0/legalcode', uri='http://www.gbif.org/dataset/6d2cfc0a-9903-40b8-802b-403398218e4a')),
 ProvenanceEntry(data=[1c01dbb9-e3ab-f9a2-06f5-228ba4b6bf7a:species/Canis lupus], provenance=Provenance(citation='California Academy of Sciences: CAS Mammalogy (MAM) https://doi.org/10.15468/dhbozg accessed via GBIF.org on 2023-02-17.', license='http://creativecommons.org/publicdomain/zero/1.0/legalcode', uri='http://www.gbif.org/dataset/6ce7290f-47f6-4046-8356-371f5b6749df')),
 ProvenanceEntry(data=[1c01dbb9-e3ab-f9a2-06f5-228ba4b6bf7a:species/Canis lupus], provenance=Provenance(citation="Díaz A, Martínez-Ortí A (2022). Las colecciones del Museu Valencià d'Història Natural. Version 2.5. Museu de Ciències Naturals de Barcelona. Occurrence dataset https://doi.org/10.15470/8oedep accessed via GBIF.org on 2023-02-17.", license='http://creativecommons.org/publicdomain/zero/1.0/legalcode', uri='http://www.gbif.org/dataset/a6f9f411-ab90-4d94-82de-f5edf8c3fa2a'))]
```

Learn more...

- See the VAT in action at the Tools Marketplace
- Visit the RDC Workshop
- https://docs.vat.gfbio.org

References

- Authmann, C., Beilschmidt, C., Drönner, J., Mattig, M., & Seeger, B. (2015). VAT: a system for visualizing, analyzing and transforming spatial data in science. Datenbank-Spektrum, 15, 175-184.
- Beilschmidt, C., Drönner, J., Mattig, M., & Seeger, B. (2023). Geo Engine: Workflow-driven Geospatial Portals for Data Science. Datenbank-Spektrum, 1-9.
- Statistisches Bundesamt (2024): Dominantes Ökosystem pro Rasterzelle 2018. Heruntergeladen am 13.03.2024 <https://oekosystematlas-ugr.destatis.de>.

ALL HANDS Conference

24.-26.04.2024 Munich

NFDI 4 BIODIVERSITY

More posters & publications can be found in the NFDI4Biodiversity Zenodo community: www.zenodo.org/communities/nfdi4biodiv

NFDI4Biodiversity is funded by the German Research Foundation (DFG) and is part of the National Research Data Infrastructure (NFDI), project number 442032008.