Geodatainfrastructure

Discovery and Access of data in EOC Geoservice using STAC

Felix Feckler, Jonas Müller, Torsten Heinen, Hendrik Zwenzner, André Twele, Markus Kunze, Jan-Karl Haug and Katharina Emde

German Aerospace Center (DLR), German Remote Sensing Data Center (DFD), Oberpfaffenhofen, D-82234 Weßling, Germany
Email: [Felix.Feckler, Jonas.Mueller, Torsten.Heinen, Hendrik.Zwenzner, Andre.Twele, Markus.Kunze, Jan-Karl.Haug, Katharina.Emde]@dlr.de

Deutsches Zentrum für Luft- und Raumfahrt

EOC Geoservice

The EOC Geoservice of the Earth Observation Center (EOC) offers discovery, visualization and download services for a selection of geodata from the German Satellite Data Archive (D-SDA) and other departments of the EOC.

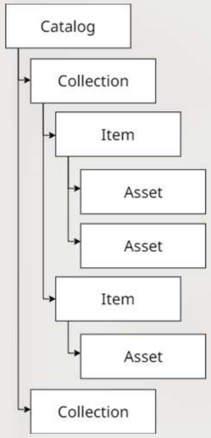
As a new web technology and an additional OGC API-based standardized access service, we add STAC to the bouquet of already existing ways to access our data.

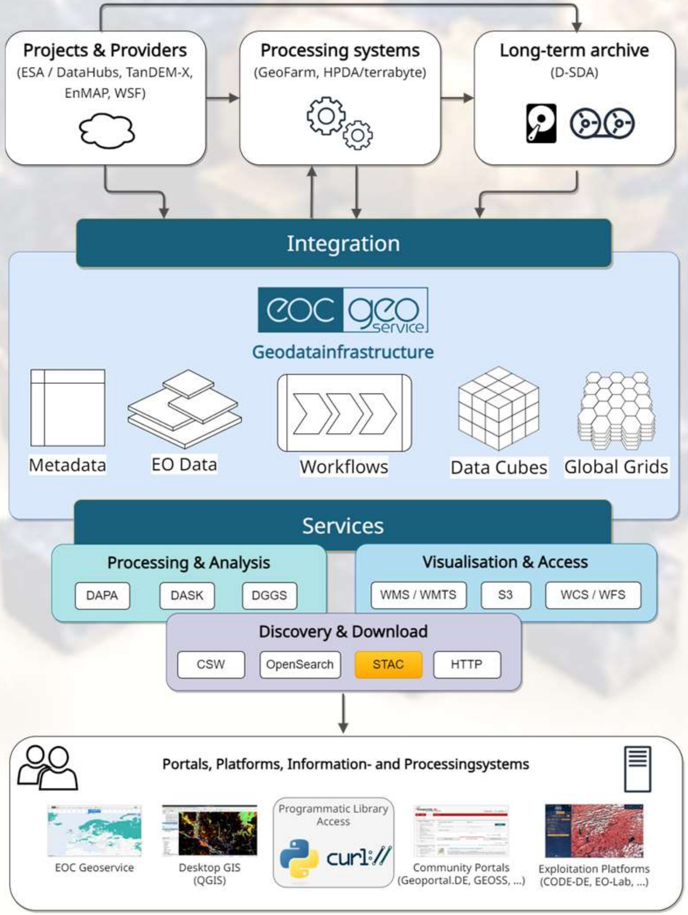
What is STAC?

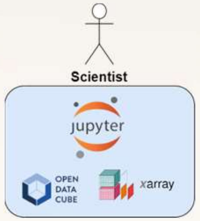
STAC SpatioTemporal Asset Catalog

STAC is a REST/OGC API based service with a common structure for describing and cataloging spatiotemporal assets. The **goal** is to expose spatiotemporal data in a **standardised** and **userfriendly** way. With a STAC API the user can **call** and **query** EO data using only lightweight metadata in JSON format and then download it to process.

STAC Components Structure







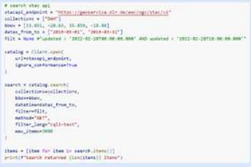
Old World

1. Use wget/FTP/GUI with the EOC download service or the EGP ordering service to download the data to your memory.
2. Doing the "ugly" preprocessing work (cropping, reprojections, etc.).
3. After these time-consuming steps and working with tons of data you don't need, the real research can begin.

New World

1. Define the **STAC API endpoint**, the **STAC Collection name** and the **spatial** and **temporal extent** for the research question.
2. Using python libraries like **pystac** and **odc** to start a query and load the wanted data into a **xarray-dataset** (data cube).
3. Now the data of the identified records will be downloaded to the local client. You can concentrate on your work. All the ugly work like reprojection and cropping is done on the server.

STAC query



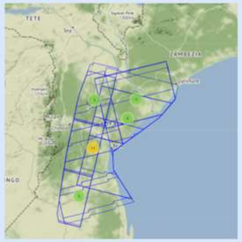
```
# search stac api
stacapi_endpoint = "https://geoservice.dlr.de/eoc/ogc/stac/v1"
collections = ["D4H"]
bbox = [33.031, -20.63, 35.059, -18.00]
dates_from_to = ["2019-03-01", "2019-03-31"]
filt = None #"updated > '2022-01-20T00:00:00.000' AND updated < '2022-02-20T10:00:00.000'"

catalog = Client.open(
    url=stacapi_endpoint,
    ignore_conformance=True
)

search = catalog.search(
    collections=collections,
    bbox=bbox,
    datetime=dates_from_to,
    filter=filt,
    method="GET",
    filter_lang="cql2-text",
    max_items=5000
)

items = [item for item in search.items()]
print(f"Search returned {len(items)} items")
```

Footprints of the STAC query



References:

1. EOC Geoservice. https://geoservice.dlr.de/web/
2. EOC Geoservice – STAC API. https://geoservice.dlr.de/eoc/ogc/stac/v1/
3. STAC: Spatio Temporal Asset Catalogs. https://stacspec.org/en
4. UKIS data-tutorials: Access Data4Human Sentinel-1 Floodmasks. https://github.com/dlr-eoc/ukis-data-tutorials/blob/main/water/access_data4human.ipynb
5. STAC API Foundation Specification. https://github.com/radiantearth/stac-api-spe