NEXTGEOSS

European Data Hub and Platform

Data Discovery

NextGEOSS User Guide

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 730329.

1.1. NextGEOSS Catalogue and data search

NextGEOSS Catalogue is based CKAN, an Open Source software. CKAN uses the Open Knowledge Foundation's Versioned Domain Model (VDM) to keep a complete history of all edits and versions of a metadata record related to a dataset. Using this feature, it is possible to look at a complete history of changes related to a dataset, and compare different revisions. Moreover, all of CKAN's core functionality (everything that can be done through the CKAN web client), is available through the CKAN API.

The CKAN API provides access to:

1. Full querying / searching (with all features of the main interface, including full-text search, querying on any attribute and faceting)
2. Full dataset information, including download links
3. Dataset listings by publisher, or by theme, etc.
4. Recent activity and additions (also available via RSS/Atom feed)
5. Statistics on dataset usage, such as number of downloads of dataset resources using the Google analytics extension
6. RDF version of the catalogue (using the rdf extension)
7. CSV & JSON dumps of entire catalogue

In addition to the read API, a write API can be provided for authorized users that allows for full update of dataset information (metadata). This enables publishers to easily integrate dataset publication with existing tools and workflows.

1.1.1. Using Catalogue search from client applications

OpenSearch is "*a collection of technologies that allow publishing of search results in a format suitable for syndication and aggregation. It is a way for websites and search engines to publish search results in a standard and accessible format.*"

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 730329.

For more information about OpenSearch, see:

- https://en.wikipedia.org/wiki/OpenSearchhttp://www.opensearch.org/Home

- http://www.opensearch.org/Home (OpenSearch 1.1 draft 6 specification and OpenSearch extensions)
- http://www.opengeospatial.org/standards/opensearchgeo (OGC standard - OpenSearch Geo and Time Extensions)
- http://www.opengeospatial.org/standards/opensearch-eo (OGC standard - OpenSearch Extension for Earth Observation)

OpenSearch is intended as an M2M (machine to machine) application programming interface (API) that provides a standardized way of 1) submitting search queries and 2) receiving results via HTTP with a special protocol. The results are returned as XML. If you have an OpenSearch client application or if you're a developer who needs a standards-based way of interacting with the data hub in your code, the OpenSearch interface will be useful. Otherwise, OpenSearch is probably not what you need and you're better off using the main web-based search interface accessible from the data hub's home page.

To use the OpenSearch interface with your client, you'll need to access the description documents first. Once your client software has parsed the description documents, you'll be able to make queries. The description documents will also tell your client what endpoints to use for search queries.

The sections below explain how and where to access the description documents, what the default parameters are, and what content you can expect in the search results.

This information is maintained by the NextGEOSS partner Viderum in the following documents:

- NextGEOSS technical note - OpenSearch on the NextGEOSS DataHub v4, Kevin Brochet-Nguyen, Viderum

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 730329.

- NextGEOSS deliverable D2.2.2 Data Discovery Guide - Version 2

It is reported here below for the reader's convenience, and is a snapshot summarised in a condensed way of the guidance status by the NextGEOSS EP-2 milestone.

Note: while the NextGEOSS DataHub OpenSearch interface might evolve and add more search features, the principle remains the same, and application developers taking care of implementing the initial phase of retrieving the OpenSearch Description documents will be able to adapt to these changes without breaking their client software code. In NextGEOSS,

The Pilot application developers are supported on Terradue Cloud Platform with the "Ellip workflows" solution (see dedicated section above) with the *opensearch-client* tool being part of the provided SDK.

1.1.1.1. Using the OpenSearch Description documents

All the default parameters are defined in the description documents. They are all standards-based.

By default, the description documents are located at /opensearch/description.xml. The osdd parameter is required and determines which description document will be returned.
/opensearch/description.xml?osdd=dataset

By default, any portal using the extension will have a dataset description document that describes how to search all the datasets on the portal using the available parameters. The "dataset" search is equivalent in scope to the standard search on portal's web frontend and via the API.
/opensearch/description.xml?osdd=record

If record view is enabled, the portal will have a record description document that describes how to access an XML representation of individual datasets in the OpenSearch Atom format.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 730329.

This provides a way to link to individual datasets without relying on search terms that may change.

The record view endpoint accepts one and only one required parameter that refers to a unique identifier (by default, this will be the identifier extra). The machine-readable description document describes the parameter in more detail.
/opensearch/description.xml?osdd=collection

If collections are enabled, the portal will have a collection description document that describes how to search for collections of datasets. This is step one in two step search, where the user first searches for a collection that meets their search criteria and then searches within the collection to find the dataset or datasets they need. The result of a collection search is a list of links to description documents for individual collections. The user selects a collection and then executes a new search using the parameters defined in the collection's description document.
/opensearch/description.xml?osdd={collection_id}

If collections are enabled, the portal will have a description document for each collection that is available. These description documents are used in step two of two-step search (see preceding paragraph). The result of a search within a specific collection is a list of datasets *belonging to that collection* that match the search parameters. Each collection description document may be unique—some collections may support parameters that others don't. For instance, one collection might support queries about sensor type (because sensor type is part of the metadata of its datasets) while another might not (because its datasets do not contain sensor type metadata, or because sensor type is irrelevant).

1.1.1.2. Using the default Search parameters

Consult a description document, and the related standards for details about the supported search parameters.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 730329.

By default, the following parameters are supported:

- opensearch:searchTerms
- opensearch:maxResults
- opensearch:startPage
- geo:box
- geo:uid
- time:start
- time:end
- referrer:source
- eo:modificationDate
- geo:geometry

1.1.1.3. Using the Search results

Search results are returned as XML Atom feeds. Atom is a syndication format.

In addition to the Atom and OpenSearch namespaces, the results use elements defined by OGC's OpenSearch Geo and Time extensions, among others.

The results are intended to comply with OGC's best practices for OpenSearch in the Earth observation field.

Default elements in Atom results feed are chosen to comply with OGC OpenSearch best practices.

For more details, consult OGC's standards here:

- http://www.opensearch.org/Specifications/OpenSearch
- http://www.opengeospatial.org/standards/
- https://tools.ietf.org/html/rfc4287 (the IETF RFC for Atom)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 730329.

1.1.2. Ingestion of metadata records into CKAN catalogues

Third-party data providers identified by the NextGEOSS Pilots as providers of Open Data resources useful for their Pilot application can be included as registered resources, accessible through the NextGEOSS DataHub.

Also, Partners in charge of supporting the Pilots (e.g. DLR, NOA, CLS/CMEMS...) with their data offering can have their data servers included as registered resources, accessible through the NextGEOSS DataHub.

To this end, ad-hoc software connectors are designed as "CKAN extensions", and implemented in the NextGEOSS DataHub, in order to harvest the Data Porvider's metadata records (note: this is an online operation, running over web service endpoints) that are describing the relevant data collections.

1.1.2.1. Using connectors as CKAN remote harvesting extensions

Within NextGEOSS, the evolvability of the CKAN Open Source software is exploited in order to create ad-hoc software connectors, designed as "CKAN extensions".
A CKAN connector is implemented according to a specification describing a web service endpoint hosting metadata (typically, a catalog endpoint) that documents the data collections and data products to be exploited by a NextGEOSS Pilot.

When ready, a connector is implemented in the NextGEOSS DataHub, and is operated in order to feed (and update) the NextGEOSS dataHub with metadata records. Such records allow search operations to select a specific part of a data collection (datasets matching search criteria such as gelolocation, creation dates, keywords...). The search reaults include the information about the datasets access endpoints, where actual datasets can be fetched by a Pilot application (especially, the data processing component of it).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 730329.

Operational Metadata harvesters tested within the NextGEOSS DataHub are currently:

- Sentinel 1, 2, 3
    - Domain: ground surface parameters, clouds, aerosols
    - Connector developped with redundant "multi-respository" harvesting, encompassing dataset sources from the Copernicus SciHub, the Hellenic Sentinel DataHub (NOA) and the German Copernicus collaborative ground segment (CODE-DE)

- MetOp/[AB]-GOME-2
    - Domain: atmospheric trace gases e.g. ozone, NO2, SO2,...
- CMEMS
    - Domain: Marine environment

The guidance for the integration of a new CKAN Connector is available from:

- https://docs.ckan.org/en/ckan-2.7.3/extensions/tutorial.html (CKAN documentation)
- https://github.com/ckan/ckanext-harvest (ckanext-harvest - Remote harvesting extension, a common harvesting framework for ckan extensions)
- NextGEOSS deliverable D2.1.1 Data Ingestion Guide - Version 2 (NextGEOSS Connectors specific aspects: harvesting from multiple repository sources, use of Dublin Core as common information model, ...)

1.1.2.2. Using metadata enrichment techniques - the ws-iTag service

iTag is an external application that can be queried during the harvesting process, to get additional metadata for datasets based on their spatial coordinates.

It is a web service meant for the semantic enhancement of Earth Observation products, i.e. the tagging of products with additional information about the covered area, regarding for example geology, water bodies, land use, population, countries, administrative units or names of major settlements, added as extra fields in the common information model of a set of metadata records.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 730329.

The guidance for the integration of the iTag service is available from the following GitHub repositories:

- https://github.com/jjrom/itag (original iTag repository)
- https://github.com/Terradue/ws-itag (Terradue's iTag repository, which has a fix for the installation done for NextGEOSS, and provides guidance for data ingestion)

The metadata enrichment mechanism relies on submitting the footprint of a dataset or resource to the ws-iTag service, which returns a list of tags that are relevant for that footprint. The metadata enrichment workflow consists in the following operations:

1. Harvest a dataset's metadata record from the source (see the section"Using connectors as CKAN remote harvesting extensions")
2. Query from within the harvester's code the ws-iTag service using the dataset's footprint
3. Update the dataset's metadata using the informations returned by ws-iTag (i.e. add the retrieved information to the harvested metadata records)
4. Store the result (enriched metadata records) in CKAN

1.1.3. Using the User feedback mechanism from Catalogue client applications

NextGEOSS promotes the use of the Community Feedback Mechanism (CFM) for the sharing of users' feedback on the resources included on the GEOSS Common Infrastructure (GCI).

User feedback includes information about a resource directly provided by users, such as abstracts, purpose of the feedback, usage, ratings, user comments, discovered issues or related publications. This is of paramount importance as it is a direct mechanism to raise users' voices and promote interaction between users and data providers. Users will be more easily engaged with GEOSS activities if they can see a real opportunity to create a community and establish social links on the DataHub around the data resources they are interested in.

The data producers may take advantage of this situation, being able to respond to users' demands, in creating new versions of the resources or answering their concerns as new feedback items (related with the previous ones).

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 730329.

The first version of the conceptual model (implemented in the alpha release, June 2017) is based on the OGC Geospatial User Feedback Conceptual Model Standard and it includes its most essential features. On the next versions of the tool, the conceptual model will be revised and consolidated by the NextGEOSS project partner UREAD-UAB (reviewing outcomes from CHARMe, Melodies and W3C Spatial Data on the Web Working Group).
In several iterations through the NextGEOSS project, the implementation will be refined, adding new elements as required in response to *co-design interactions*, and also to the revision and consolidation of the conceptual model.

Regarding the implementation, a server side, a javascript html tool (client) and a service Restful API have been developed. Currently the 3 items follow the conceptual model described, and use an specific protocol that will be, at the end of the project, submitted to the OGC to be approved as an implementation standard of the GUF standard. The server implementation is currently included in the NiMMbus Cloud (it will be deployed in the EGI Federated Cloud as part of the NextGEOSS Activities) and allows to describe the feedback for any element stored in or referenced by the system.

The online guidance for the integration of the User Feedback mechanism is available from:

- https://github.com/joanma747/nimmbus (general information about the GUF from the nimmbus GitHub repository)
- https://github.com/joanma747/nimmbus/tree/master/GUF_integration (instructions to integrate the GUF)

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 730329.