Deliverable 7.2

Finished testing the system and identifying problems

Authors and affiliation:
Jean-Baptiste Roquencourt BRGM,
Eric Lecaude BRGM,
Jean Goncalves BRGM,
Andrej Vihtelič GeoZS,
Blaž Bahar GeoZS,
Jernej Bavdek GeoZS,
Maks Šinigoj GeoZS
Lucie Kondrová CGS,
Pavla Kramolišová CGS,
Miguel Ángel Alarcón IGME,
Héctor Sánchez IGME
Martin Hansen GEUS,
Bjarni Pjetursson GEUS,
Marianne B. Wiese GEUS,
Jonas Thyregod GEUS,
Viktor Rasmussen GEUS

[BENEFICIARY]

E-mail of lead author:
jb.roquencourt@brgm.fr

Version: 02-07-2020

This report is part of a project that has received funding by the European Union's Horizon 2020 research and innovation programme under grant agreement number 731166.

Deliverable Data		
Deliverable number	D7.2	
Dissemination level	Public	
Deliverable name	Finished testing the system and identifying problem	
Work package	WP7, Developments (central)	
Lead WP/Deliverable beneficiary	GeoZS	
Deliverable status		
Submitted (Author(s))	02/07/2020	Jean-Baptiste Roquencourt
Verified (WP leader)	15/07/2020	Andrej Vihtelič
Approved (Coordinator)	21/07/2020	Jørgen Tulstrup

GENERAL INTRODUCTION

This report describes the state of the European Geological Data Infrastructure (EGDI) tests that the GeoERA Information Platform Project (GIP-P) has been able to achieve. It also paves the way to solve the problems identified as well as improving the testing framework.

From the user perspective the tests are based on the requirement from D2.3.2.

This deliverable was not able to achieve the finalization of the tests because its due date in the middle of the development delivery. Therefore the scope of the document will be limited to the already functionalities that can be accessible.

As such this document will have a new version at the end of development to increase its coverage on the requirements from D2.3.2.

Standardisation was also a late communication process engaged with the GSPs, therefore the second version of the document will introduce a testing framework for better standardisation based on D3.3, and possible solution to issues based on D3.2.2.

TABLE OF CONTENTS

1.DEFINITIONS

Application Programming Interface (API): a computing interface to a software module or a system, that defines how other modules or systems can use it.

Functionality: the range of operations that can be run on a computer or other electronic system.

GeoERA: Establishing the European Geological Surveys Research Area to deliver a Geological Service for Europe.

GIP-P: GeoERA Information Platform Project.

GSP: GeoERA Scientific Project. The 14 scientific projects of the GeoERA programme.

Metadata: data that provides information about spatial and non-spatial data (e.g., purpose of the data, time of creation, authors, etc.)

Module: an application or software that is involved in serving product**.**

Product: any deliverable generated by a GeoERA project that will be available via EGDI. Projects will deliver 4 types of products.

Project vocabulary: collections of terms with short descriptions, bibliographic citations and links to unstructured web contents used to define scientific parameters and concepts.

2.INTRODUCTION

This present deliverable is a milestone to the actual finished testing the system. However, we could even argue that testing is an activity never finished, as we want to continuously improve the system in regards to user requirements while maintaining its robustness.

Our work so far was, first, to assess the scope of testing the system. The EGDI platform is composed of several modules. How should we test them and should we test all of them? What kind of test since every module is different and managed by teams of different knowledge? How to maintain a coherent vision of the test?
Hence we defined a testing strategy based on the existing test, spread it across the team and defined the potential sequel.

Then based on that shared testing strategy, we focused our attention on specific modules of the system:

a) Web GIS
b) Search System
c) 3D viewer
d) Document repository search system
e) Metadata catalogue
f) Harvesting system
g) Vocabulary tool
h) Monitoring system

Finally, having reviewed every chosen module according to the strategy, this document gives a global and coherent vision on the actual development of the system. We also proposed the sequel of the present deliverable to refine the testing of the system by the end of the project.

3.IDENTIFYING TESTING STRATEGY

3.1 What is testing and why testing

The goal of software testing is to verify the behaviour of an application against multiple criteria. A comparison could be made with a car: either you assess its look & feel, its User Experience, its robustness and so on.
The testing coverage greatly depends on a matter of fact: criticality of your application in your daily business, associated budget, time for executing the test.

Testing is often seen as a hindrance to the quickness of the development. However the consequences can be heavy. Indeed lately a certain amount of personal information have been leaked from poor secured web site[1]. In the context of GDPR such a leakage can be disastrous for the company finance[2]. At a smaller scale, the later you test, the more complex it is too find the root cause of issues, resulting in a loss of time.

Yes testing requires time, so the development team has too find a trade-off between the consequences of not testing everything, and dealing with potential critical issues.

There are over 150 ways of testing[3], however the most common can be found in the following picture:

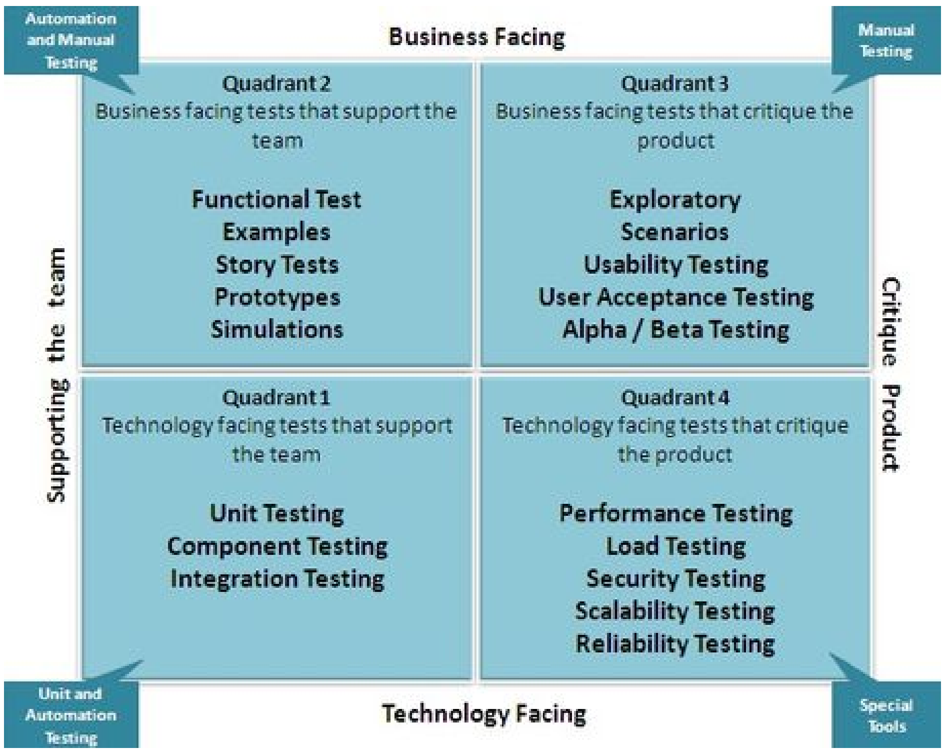


Figure 1: Software testing quadrant

[1] https://digitalguardian.com/blog/top-10-biggest-healthcare-data-breaches-all-time
[2] https://www.privacyaffairs.com/gdpr-fines/
[3] https://www.guru99.com/types-of-software-testing.html

3.2 Define the strategy

As introduced, testing is really important for the quality of the deliverables. However a too complicated quality process can hinder the respect of deadlines. For that reason we defined the strategy with the following constraints:

- Test what is interesting to test

We should not test for example a database connection that others have done for us.

- Adapt to the skills of the team

GIP-P is composed of people with a wide range of knowledge from juniors to experts with their own development culture. We will focus on developing a common approach to testing. This will build a common knowledge base and mutual understanding, facilitating collaboration.

- Adapt to the complexity of the product

The EGDI platform is composed of multiple modules. As these modules interact with each other with contract interface or API, we need to define a standard vision and understanding of every module.

- Adapt to the international team

Each module is usually developed by a core team coming from one organization. This is effective, in giving every team a tighten scope of responsibilities. As such, the strategy should not be disruptive for these organization.

- Adapt to the development agenda

We are in the middle of the development phase meaning that some functionalities and products are not yet available to be technically tested or user reviewed. We need to track them so that they will be part of the testing delivery framework.

- Adapt to the GIP-P standardization implementation process maturity

The communication with GSP in regard to the implementation of the standard is still at its early stage. Therefore, we will not take into account D3.2.2 and D3.3 for now.

- Adapt to availabilities

We don't want to have a shiny testing framework that would have consumed too much human resource time which could not be followed through. Therefore, the strategy will identify and focus on activities that will be easy to implement and could be automated (Figure 2: Automated tests).

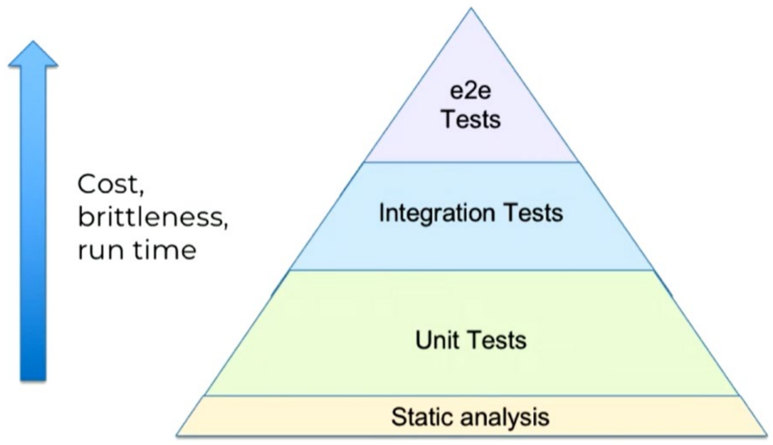


Figure 2: Automated tests

The strategy needs to secure an MVP framework for testing and give the GIP-P and EGDI platform a strong basis for the follow-up of the project and the future of the EGDI involved teams based on everyone's knowledge.

Our assumption was that there was a lot of existing automated tests in the EGDI development teams. That was not the case. Therefore, we reviewed the Software Testing Quadrant, enlightened with the following approach: the strategy will as much as possible follow the KISS principle[4], and simplified to these 2 points:
a) The most important thing for the GIP-P is the satisfaction of the users, the so-called User Experience.
b) to improve the User Experience, a lot of technical tests can be done to facilitate the development process.

3.2.1 User review

The users are the one we are working for, therefore maintaining a close relationship is of the essence. In the GIP Project, this relationship has been put into practice from the very beginning and is explained in Figure 2: Work Packages and their relationships.

[4] https://www.interaction-design.org/literature/article/kiss-keep-it-simple-stupid-a-design-principle

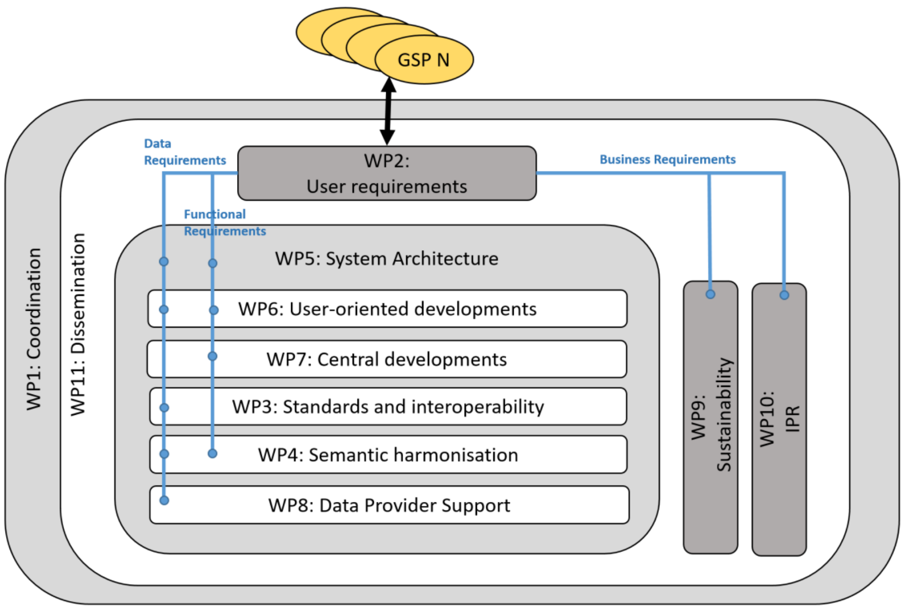


Figure 3: Work Packages and their relationships

3.2.1.1 Scheduling

As a development team, we want to avoid the tunnel effect[5]. And as this deliverable marks a milestone in the development process, we will review the latest feedback from the user. Since every module has its own timeline, and in respect to the user requirements, the team may not have had the time or the capability yet to involve the users. This should be done sooner rather than later and in a recurrent manner.

3.2.1.2 Users

As written above, the EGDI platform consist of multiple modules targeted at different kind of users. The users in the context of GeoERA are first coming from the GSPs, however some modules have been existing before GeoERA and have already established links with their users. Therefore the teams have been or will be in touch with all these users for them to review their modules.

3.2.2 Prototypes

Prototypes are used to demonstrate capabilities and to facilitate the dialog with users. WP5 has identified prototypes for Data Architecture and Service Architecture. This is out the scope of this present document version.

3.2.3 Static code analysis

As we want to improve the quality of the code delivered, we will start from the simplest automating test to put in place shown in Figure 2 : static code analysis.

[5] https://www.wethetalent.co/data-robots-ai/the-tunnel-effect-and-the-abstruse-trap/

Static analysis is best described “as a method of debugging by automatically examining source code before a program is run”[6]. It allows the detection at early stage of inefficient code, and potential security issues.

3.2.4 Unit tests

Unit tests are used to ensure that the code will work as expected by the specifications. They ensure the proper functioning of each of the sub-parts of the program (a method, a service, ...).
This practice helps to discover faults when programming a feature and, above all, to automatically check that malfunctions have not been introduced when changes are made to the code of an application.
In order to be as relevant as possible, automated tests must include unit tests and not only integration tests:

a) A unit test controls an elementary fragment composing a functionality. It is therefore more targeted and allows to locate potential errors more easily.
b) Unit tests should not test frameworks. We know that others have done it before us, and they should test isolated layers from each other and then, through integration tests, the whole call chain through the architecture.

Particular attention is paid to the testing of business rules (service layer) and to the mapping of data objects (service layer but potentially also the controller).

3.2.5 Scenarios: Integration and End to End tests

Although different in their scope, as up to now integration and End-to-End (e2e) tests have been carried out for each module and also on the level of the whole EGDI platform manually. Although we could work on their automation, we think it is too early to work on this. These tests are being done regularly, and have improved the overall coherence of the delivered software application or website.

3.2.6 Security tests

We will focus on black box security testing with penetration tools.
The realization of such a test needs to be held with great care. Laws are written to protect the website owner therefore any attempt to misuse it or breaking attempt can be severely punished. Therefore before performing the test you need to be granted the right to do so.
Therefore the strategy would be to: a) transfer the knowledge on security testing to each module teams, b) each module team performs the test, c) report the result to the project leader and d) automatize the test.

3.2.7 FAIR assessment

Being FAIR is a journey, and we should assess the FAIRness of EGDI. There are already existing methodologies for FAIR assessment[7]. As EGDI is part of the TCS Geology of EPOS ERIC, we could benefit from EPOS active involvement in ENVRI-

[6] https://www.perforce.com/blog/sca/what-static-analysis
[7] https://www.rd-alliance.org/sites/default/files/Ms%20MAIN%20TEXT-%20R.%20David-17jan2020%281%29.pdf

FAIR. EPOS has proposed a mechanism to FAIR assessment[8]. From that experience, we know that FAIR Assessment requires time and we want a less complex methodology Thus our pragmatic solution on this matter is:

a) Based on the fact that the EGDI platform is already compliant, to a certain degree to Inspire and Standard like OGC which are FAIR by design.
b) And as GIP-P WP3 already presented to WP7.2 existing software for testing the INSPIRE and OGC capabilities of a service we will focus on these tool first.

3.2.8 Performance testing

We envisioned the following process:

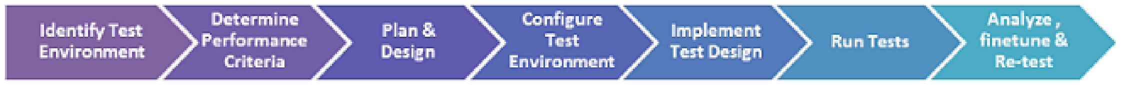

Identify Test Environment
Determine Performance Criteria
Plan & Design
Configure Test Environment
Implement Test Design
Run Tests
Analyze , finetune & Re-test

Figure 4: Performance testing process[9]

However while identifying the test environment we needed to understand the overall architecture. We found that there was still some discrepancies on the environment documentation. We also struggled to define the KPIs.

3.2.8.1 Technical architecture

Each module teams were asked to provide a common level of architecture description. We agreed on the following information:

a) Architecture diagram
b) Hardware component
c) Software component

3.2.8.2 Metrics or KPI

So far we were not able to identify written KPIs. The reason being that the existing modules are already working for their users, and overall the users seem satisfied. However, we envision that there is a certain level of information that needs to be shared, so that the new modules cope with the actual usage of the platform.
This deliverable will first capture the existing metrics for every module with a basic layer composed of time base metrics:

a) The number of users per day (hour, year, etc.)
b) The number of requests per day (hour, year, etc.)

For a better caption of the metrics, this deliverable suggests to adopt a strategy for the harmonization of metrics. It is suggested that WP7 in cooperation with WP5 and WP6 takes the lead on this. This deliverable suggests to start using with web analytics such as Matomo[10], and improve it with log file indexes: Elasticsearch [11]or Graylog[12]. These software are the de facto open source software at the deliverable writing time.

[8] https://www.frontiersin.org/articles/10.3389/feart.2020.00003/full
[9] https://www.guru99.com/images/performance_testing_process.png
[10] https://matomo.org/
[11] https://www.elastic.co/
[12] https://www.graylog.org/

3.2.9 Synthesis

The strategy can be summarized by with the addition of static code analysis and FAIR assessment.

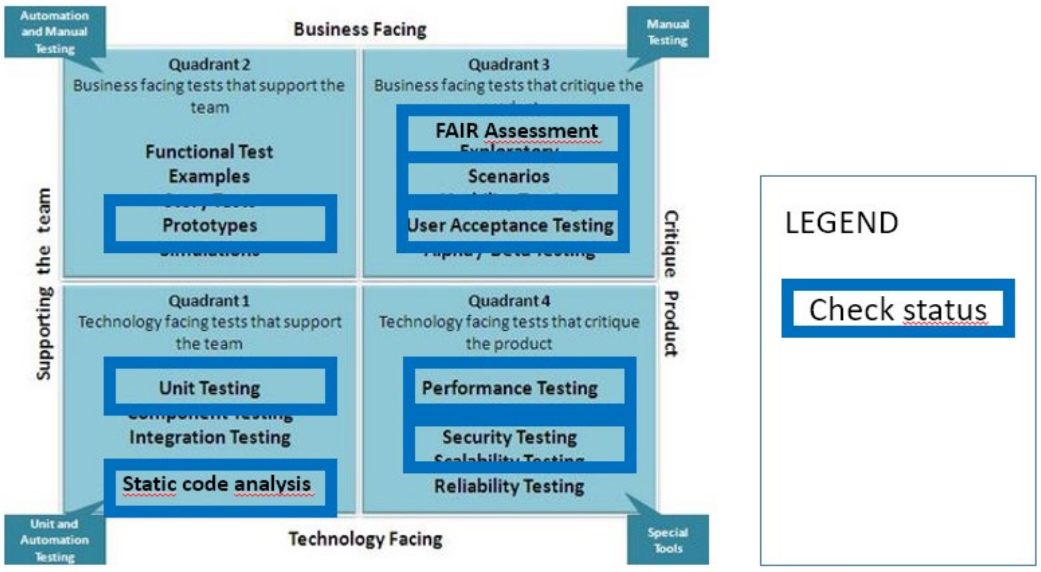


Figure 5: Defined strategy

4. THE EGDI TECHNICAL LANDSCAPE PROVIDES A GLOBAL VISION FOR THE EGDI ARCHITECTURE

The chapter will introduce the modules that are used by the end users to visualize data. We will be giving a detailed representation of relevant modules, while the description of the administration and monitoring modules will be less detailed.

4.1 Web GIS

4.1.1 Introduction

The EGDI Web-GIS System disseminates data products and data sets as online interactive maps including various tools to further perform data analysis. The most current version is available as embedded maps on the project pages of https://geoera.eu and at http://www.europe-geology.eu/

From a user-perspective, the interactive maps comprise the following key functionality:

- Browse index of all available contents in EGDI
- Gather more information from legends and metadata records
- Subset selection using attribute filters and map interactions (mouse and touch)
- Adding external map sources (WMS)
- Attribute details on features (data analysis step 1)
- Linking to custom data viewers on feature and data set level (data analysis step 3)
- Downloading of raw data for local work (data analysis step 3)

In addition, the maps offer functionalities expected from state-of-the-art online maps:

- Support for multiple screen sizes
- Map navigation using location search
- Map navigation using map interactions (zoom and pan)
- Measuring tool for paths (m) and polygons (m2)
- Change layer drawing order
- Zoom to current location
- Permanent link option
- Fullscreen option
- Rotate north option

The system is a full-stack system using client-side logic (browser), server-side logic, and data storage.

Client-side logic	Interactive web page with embedded logic using common browser modules like jQuery, Bootstrap and OpenLayers. OpenLayers takes care of orchestrating map layer contents from various sources into a web-GIS. In addition, several

	libraries are built in the scope of EGDI to allow for more functionality.
Server-side logic	The EGDI server runs a Java Tomee application server with an EGDI application for serving settings and contents necessary to fill the web-GIS. The application makes use of Mapserver seamlessly translate layer configurations into map services.
Storage	The EGDI application relies on a PostgreSQL database to store and retrieve both map configurations and contents. This enables administrators to manage both data products and their configuration in one place.

4.1.2 Architecture

4.1.2.1 Production environment

The system server runs Ubuntu Linux with 4 cores and 16 GB RAM. Installed are:

a) Java SDK/Payara latest stable
b) PostgreSQL latest stable
c) Tomcat latest stable
d) Mapserver binaries latest stable including bridge to Java Mapscript
e) Custom build web application "egdi.war"

The most current installation instructions can be found in GitLab:

https://geusgitlab.geus.dk/egdi/egdi/-/tree/master/docs > 00_install_egdi_on_linux.txt

Other useful instructions can be found in the parent folder.

4.1.2.2 Test environment

The EGDI Web-GIS test and production environment runs on identical hardware as the production environment. At regular intervals the test server is overwritten with a fresh image copy from production.

4.1.2.3 Architecture diagram

The EGDI Web-GIS fits into the overall architecture described in this:

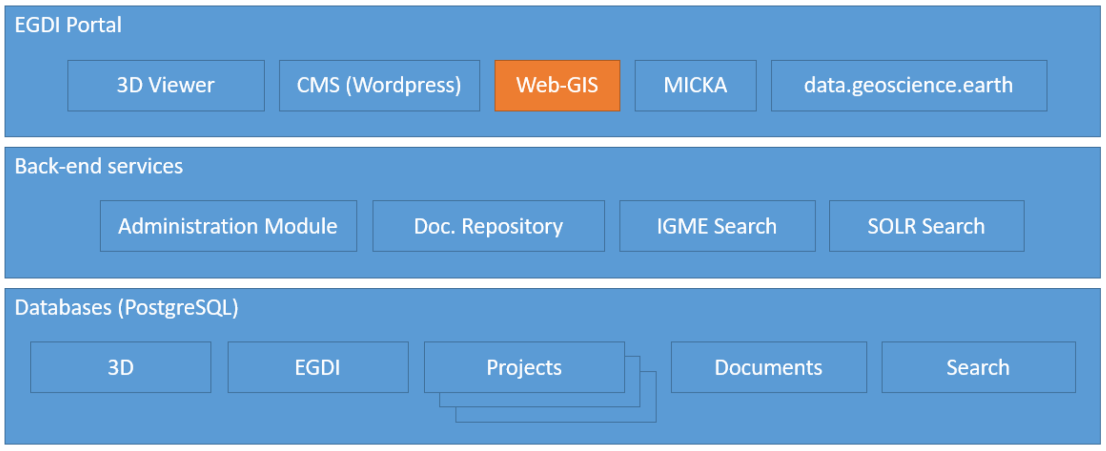
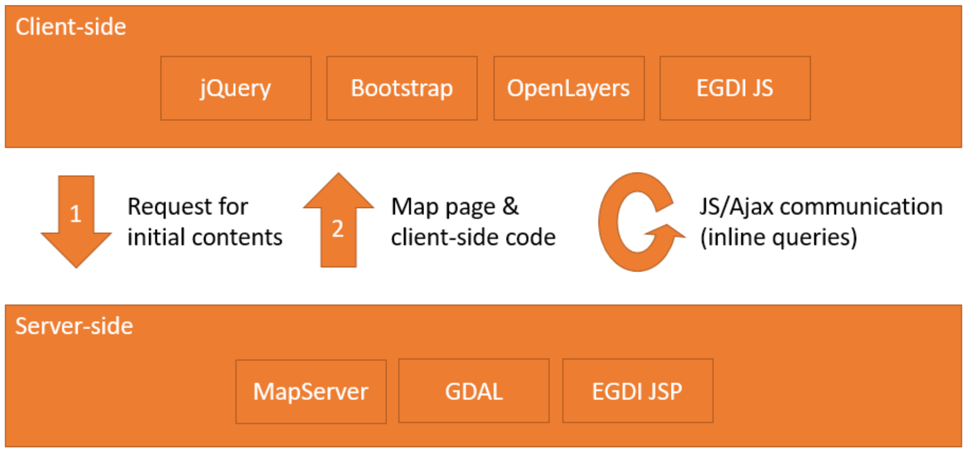


Figure 6: Web-GIS fits in overall architecture The Web-GIS is comprised of the following components:



Client-side
jQuery
Bootstrap
OpenLayers
EGDI JS
1
Request for initial contents
2
Map page & client-side code
JS/Ajax communication (inline queries)
Server-side
MapServer
GDAL
EGDI JSP

Figure 7: Web-GIS is comprised of the following components

4.1.3 Metrics

Offering Web-GIS to users involves pushing functionality to the client (browser) and generating maps, when the user interacts with the map. Pushing functionality to the user requires very little server-side resource. The primary Web GIS bottleneck is by far the server-side generation of map images. Therefore, the metrics will focus on this process.

4.1.3.1 Number of users

The system rarely has more than 20 users browsing the portal at the same time. The number of simultaneous users on the Web-GIS is even lower and peaks during working hours. We could have considerable spikes in usage like BRGM experienced with the publication of the OneGeology Europe map, where news articles prompted a huge interest from the public to browse for data. These events will require a completely different setup with cached map tiles (pre-compiled images) and a simpler user-interface. It is important to ensure that the planning of such events is coordinated with the operational team for EGDI.

4.1.3.2 Number of requests

EGDI is currently scaled to support around 10 map generations in parallel. Some map generations take below 100ms. Other map generations could be well above 20 seconds.

4.1.4 Envisioned scenario for performance testing

4.1.4.1 Stage 1

Empty browser cache
Go to http://www.europe-geology.eu

4.1.4.2 Stage 2

Click the "All Content" entry from the top menu
A map of Europe should be displayed

4.1.4.3 Stage 3

Activate the first layer in the index "euRare occurrences"
Content should appear on the map

4.1.4.4 Stage 4

Zoom to Greece and click a feature on the map
A text list should appear at the bottom of the map

4.1.5 Scalability issues

The current system is built upon GEUS' own Web GIS and has been running on the EGDI platform since 2016 with little modification needed. EGDI web-GIS currently has around 400 map page requests (https://data.geus.dk/egdi) per day with a total of around 6000 related file requests per day (map images, legends, metadata, images etc.).

Here is a typical week of activity (please note, the COVID19 situation may have influenced stats):

Date	Total	images etc.	proxy	legends etc.	map images	wfs	wms	Core requests
12-05-2020	6.354	3.357	313	238	1.909	42	80	415
13-05-2020	3.699	2.363	106	134	749	12	70	265
13-05-2020	5.959	2.690	568	212	2.141	20	63	265
15-05-2020	7.288	3.415	672	266	2.295	78	144	418
15-05-2020	5.426	3.154	356	189	1.222	157	48	300

16-05-2020	3.163	1.586	209	134	1.089	0	7	138

It is not expected that the use of the platform for GeoERA will give a large increase in the number of users.

Interactive web maps with high levels of complex data and user requirements are difficult to scale without sacrificing agility. They are also difficult to fit into a standard service-level-agreement due to the many possibilities of use offered. The main issues will be addressed in this chapter.

4.1.5.1 Horizontal scalability via load-balancing

The current system runs on a single Linux node with four cores. This setup is very agile in terms of deployment and management. This means that very little effort is required in daily handling and management of the system. The system can be easily scaled via load balancing using several identical servers/dockers. This will be a linear scaling meaning that two servers will double the throughput. However, it will nearly also double the work needed to deploy new software and manage the setup in case of problems thereby harming agility.

Managing the setup is no easy task because of the many processes and modules involved. Analysing the core problem through several servers would require extra work if using load-balancing.

Currently, we can handle 5-10 simultaneous processes per core. Depending on data and request complexity, a process can last anywhere between 0,1 to 30+ seconds. This might sound like a serious problem, but the current load of EGDI with just a single server can cope. The bottleneck is processing power, due to the math involved in translating GIS-data to raster images. We use Mapserver to read subsets of a data from the database, transform it into the correct projection and translate each feature using styling rules into a raster image.

We can expect higher load as GeoERA matures with more user activity and more data products. In that case, it is advisable to introduce load-balancing gradually starting with perhaps 2-3 servers/dockers in parallel. The overhead would be acceptable, with the added benefit of being able to upgrade the nodes one at a time with little or no downtime.

4.1.5.2 Vertical scalability via raster image caching and user constraints

Huge, static data sets that rarely change and where users just want to show the complete data on a map are good candidates for raster caching. It requires extra work to establish and data are “dead” in the sense that further data analysis is not possible. On the other hand, you can zoom and pan through a huge data set with little or no lag. EGDI Web-GIS is mainly built for working with the data and therefore currently does not use raster image caching. A few geological maps are served externally through ArcGIS or

Geoserver cache (e.g. the surface geological maps) and are available in EGDI Web-GIS through proxy.

We can expect higher load in the future and caching could be a solution. The complexity of managing and serving the data would increase.

Another solution for administering high load could be to introduce user constraints like maximum requests per day per IP-address, limiting the amount of data available in each response or require user id in each request. This approach will negatively impact user experience and increase resources needed for user support. On the positive side, it can result in more collaboration with end-users on establishing a robust and stable platform.

4.1.5.3 Software

EGDI Web-GIS runs server-side as a Java web application. All functionality and content is served from the EGDI server through a Java web server to the client. For Web-GIS, the client is a browser, where the Web-GIS functionality activates client-side code to offer interactivity directly in the browser through the use of JavaScript.

The interactive map requests data from the server when the user pans, zooms or clicks the map. Because the data has spatial contents and should be served as map images, the freeware Mapserver library is used to complement Java.

4.1.5.4 Hardware elements

EGDI Web-GIS runs on the production server GEUSEGDI01 with no additional hardware requirements.

4.1.6 User review

The EGDI Web-GIS was thoroughly tested during implementation in 2016. Development since then has been incrementally added with ad-hoc testing.

4.1.7 Performance testing

The EGDI Web-GIS has been in production since mid-2016 and is known to perform linear according to the hardware resources supplied (CPU cores). Acknowledging this bottleneck, the EGDI development team introduced measures to push more work to the client. For example, many maps are hosted by external services (WMS). The Web-GIS will make a direct connection to the external service without involving the EGDI back-end.

Tests have shown that around five simultaneous map requests can be handled server-side per CPU. We are currently on a server with two cores on a virtual environment.

4.1.8 Planned improvements

It is planned to have the system enrolled in CI/CD using GitLab together with its related application "egdiadmin".

Due to the abovementioned problems running stable map services in large scale, it is planned to have map services run on separate server/Docker installation. This will isolate potential downtime to services only and narrow down the potential erroneous modules when doing maintenance.

Other improvements include a revitalized user interface with less functionality cluttering the map. A top menu bar will replace many of the embedded tool buttons and layer-specific functionality will for clarity be gathered in a drop-down menu.

4.2 Development of a search system

4.2.1 Introduction

The Search System is a multilingual web system to let users:

- Discover resources, that is, discover useful or valuable geoscientific information produced in the GeoERA projects and the one already available through the European Geological Data Infrastructure (EGDI).
- Access the available information, services or functionality for a resource through their distributions. Distributions are locations for on-line access related to a resource: the resource in a certain format, web pages, applications or services related to the resource and any other information directly related to the resource
- Search inside some resources, supplying subsets of records in a database, documents in a document repository or concepts in a project vocabulary (selected features type depends on the resource type).

The back-end development platform mainly used for the Search System is .NET Core 3.1. This technology is a cross-platform version of .NET for building websites, services, and console apps that aims to supply best performance and ideal scalability. It is a recent technology, specially designed to improve all these aspects. Several applications have already been developed using .NET Core at IGME-ES and the experience has been really satisfactory.

The Search System application has been deployed in two different environments:

- Internet Information Server (IIS) on Windows Server 2016.
- Apache on Linux (Ubuntu 18.04).

4.2.2 Architecture

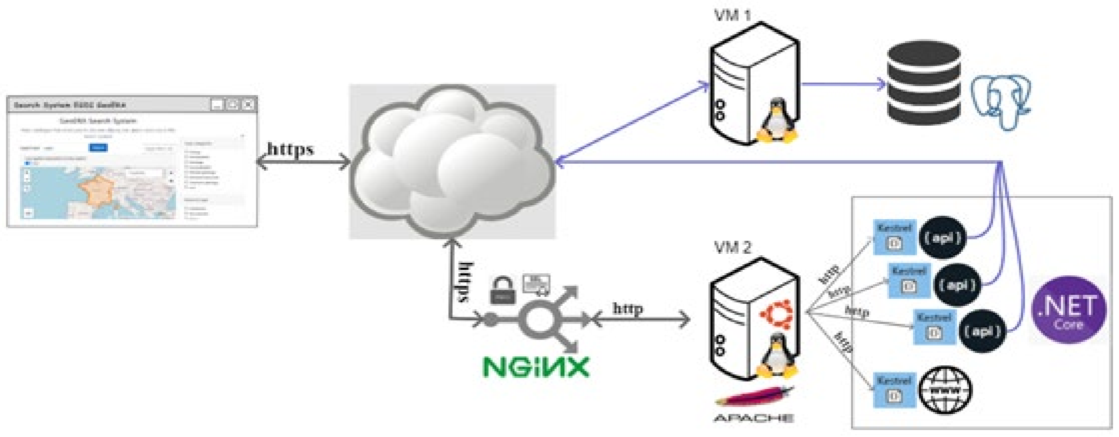


Figure 8: Search system architectureIGME-ES has deployed the Search System in two environments, IIS with Windows Server and Apache+Kestrel on Linux. The following diagram shows the Linux publication, as we believe the final production environment will be similar to that one. The specific details of the production environment are currently unknown to us.

4.2.2.1 Production environment

The production environment has not been fully defined yet. The IGME-ES' development environment has been established based on what is indicated within WP6 and 7 (https://geusgitlab.geus.dk/egdi/technical-documents/-/blob/master/UsedSoftware.md). An overview of what the final production environment could be is shown in the next section.

4.2.2.2 Architecture proposal

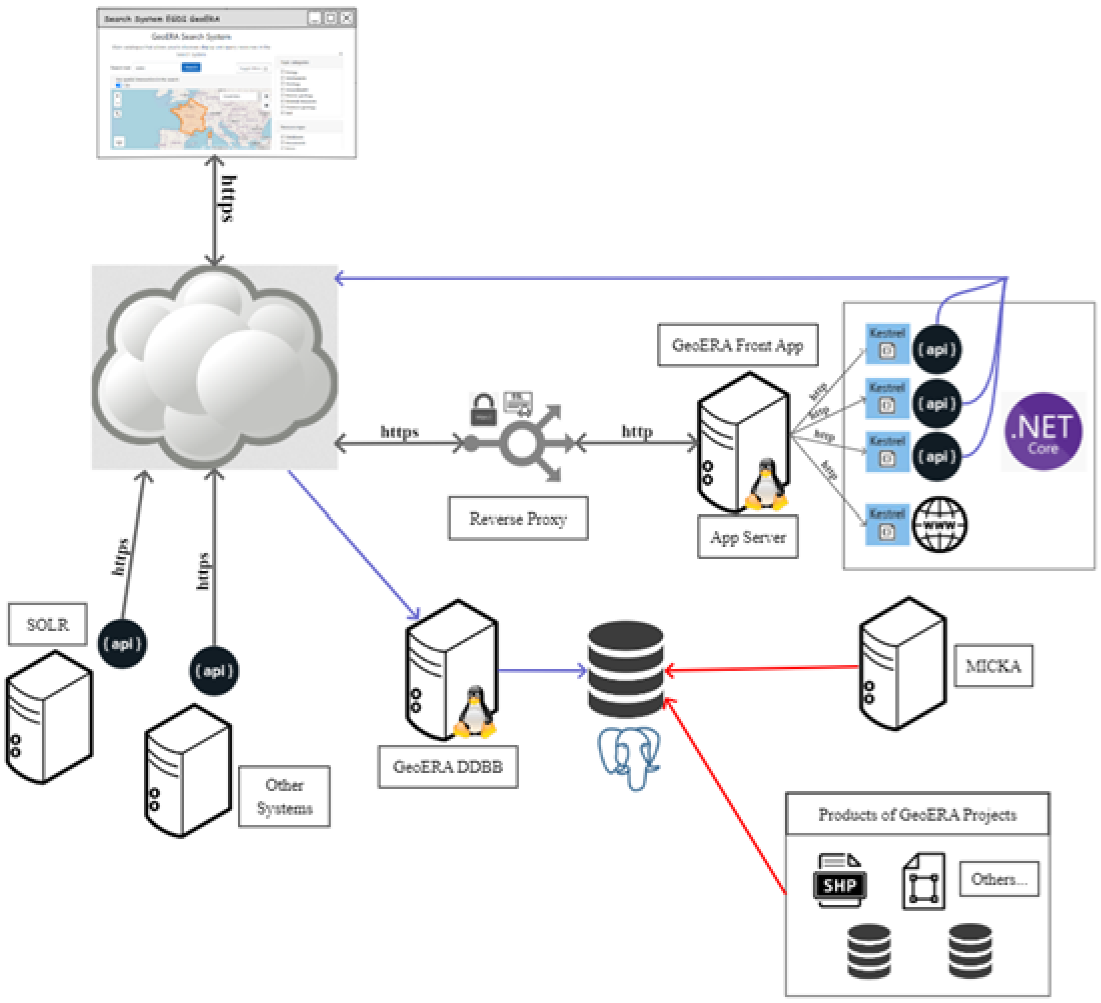


Figure 9: Search system architecture proposal

4.2.2.3 Software

https://geusgitlab.geus.dk/egdi/technical-documents/-/blob/master/UsedSoftware.md

Module	Type	Version
	Database Manager	PostgreSQL 10.10 + PostGIS 2.4.4
Apache HTTP SERVER	Http Server	Apache v 2.4.x
NGINX	Reverse Proxy	1.10.3
.NET Core	Development framework	Microsoft .NET Core v3.1
VM 1	Operating System	Linux – Debian 9.11 Strech
VM 2	Operating System	Linux – Ubuntu 18.04LTS

4.2.2.4 CPU and RAM

VM	CPUs	Memory
Virtual Machine 1	2	1024 Mb
Virtual Machine 2	4	8192 Mb

4.2.2.5 Software disk space usage

Software element	Space disk (Windows)	Storage technology (Windows)	Space disk (Linux)	Storage technology (Linux)
Database (Metadata, thesarurus, configuration, spatial data and Feature Distribution examples).	767 Mb	VMWare Virtual SCSI - NTFS	767 Mb	Hyper-v Virtual IDE – Ext4
MVC application	40 Mb	VMWare Virtual SCSI - NTFS	39 Mb	Hyper-v Virtual IDE – Ext4

Suggester API	80 Mb	VMWare Virtual SCSI - NTFS	42 Mb	Hyper-v Virtual IDE – Ext4
String processing API	80 Mb	VMWare Virtual SCSI - NTFS	42 Mb	Hyper-v Virtual IDE – Ext4
Spatial API	24 Mb	VMWare Virtual SCSI - NTFS	26 Mb	Hyper-v Virtual IDE – Ext4
Spatial RCL	2 Mb	VMWare Virtual SCSI - NTFS	2 Mb	Hyper-v Virtual IDE – Ext4
Feature Distribution API for PostgreSQL	40 Mb	VMWare Virtual SCSI - NTFS	37 Mb	Hyper-v Virtual IDE – Ext4
Pool of Feature Distribution APIs	Unknown. Will depend on the implementation.			

4.2.2.6 Performance: CPU RAM disk

Link to gitlab

Server/virtual machine name	CPU	RAM	Disk

4.2.3 Metrics

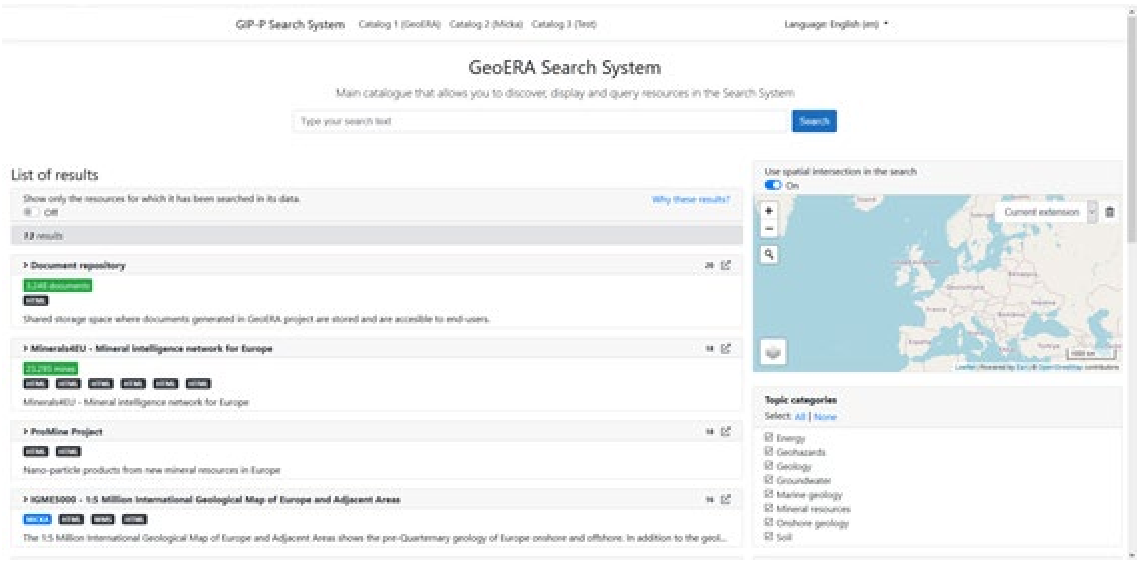
Currently the system is not in production and therefore no metrics are available.

4.2.4 Envisioned scenario for performance testing

Empty your browser's cache.

4.2.4.1 Stage 1

Go to the following web address https://info.igme.es/SearchSystem/v02/en/GeoERA (it gives access to the publication made in IIS)


GIP-P Search System Catalog 1 (GeoERA) Catalog 2 (Micka) Catalog 3 (Test) Language: English (en)

GeoERA Search System

Main catalogue that allows you to discover, display and query resources in the Search System

Type your search text Search

List of results

Show only the resources for which it has been searched in its data. Off Why these results?

12 results

Document repository 26
3,248 documents
HTML
Shared storage space where documents generated in GeoERA project are stored and are accesible to end-users.

Minerals4EU - Mineral intelligence network for Europe 18
23,293 mines
HTML HTML HTML HTML HTML HTML
Minerals4EU - Mineral intelligence network for Europe

ProMine Project 18
HTML HTML
Nano-particle products from new mineral resources in Europe

IGME5000 - 1:5 Million International Geological Map of Europe and Adjacent Areas 16
MICKA HTML WMS HTML
The 1:5 Million International Geological Map of Europe and Adjacent Areas shows the pre-Quaternary geology of Europe onshore and offshore. In addition to the geol...

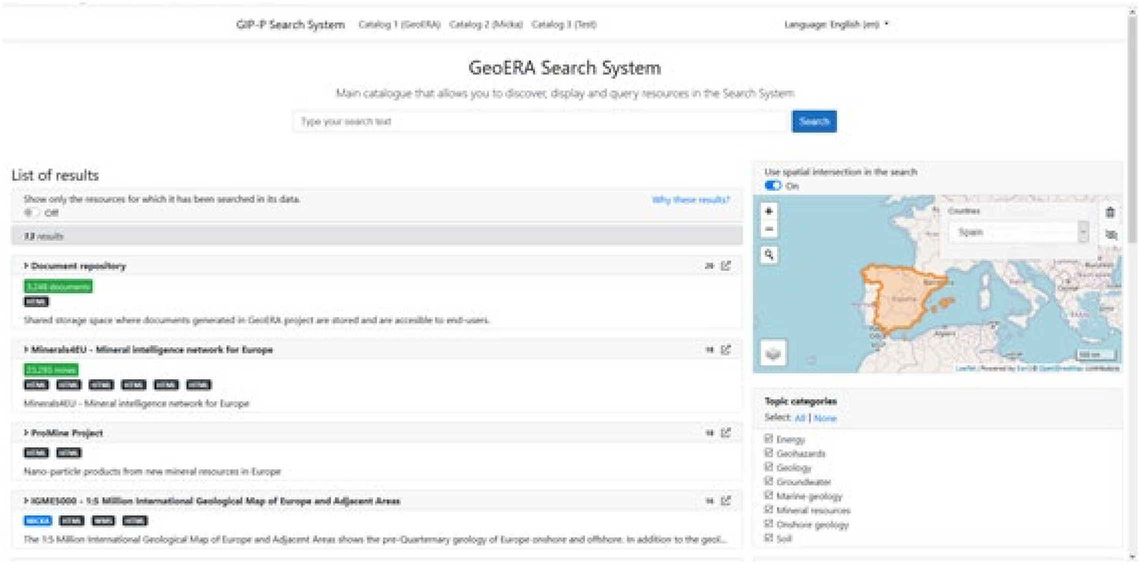
Use spatial intersection in the search
On
Current extension

Topic categories
Select: All | None
☑ Energy
☑ Geohazards
☑ Geology
☑ Groundwater
☑ Marine geology
☑ Mineral resources
☑ Onshore geology
☑ Soil

Figure 10: https://info.igme.es/SearchSystem/v02/en/GeoERA

4.2.4.2 Stage 2

Go to the Spatial Selector (map in the right side). In dropdown showing the default value "Current extension", select "Countries" and then "Spain".


GIP-P Search System Catalog 1 (GeoERA) Catalog 2 (Micka) Catalog 3 (Test) Language: English (en)

GeoERA Search System

Main catalogue that allows you to discover, display and query resources in the Search System

Type your search text Search

List of results

Show only the resources for which it has been searched in its data. Off Why these results?

12 results

Document repository 26
3,248 documents
HTML
Shared storage space where documents generated in GeoERA project are stored and are accesible to end-users.

Minerals4EU - Mineral intelligence network for Europe 18
23,293 mines
HTML HTML HTML HTML HTML HTML
Minerals4EU - Mineral intelligence network for Europe

ProMine Project 18
HTML HTML
Nano-particle products from new mineral resources in Europe

IGME5000 - 1:5 Million International Geological Map of Europe and Adjacent Areas 16
MICKA HTML WMS HTML
The 1:5 Million International Geological Map of Europe and Adjacent Areas shows the pre-Quaternary geology of Europe onshore and offshore. In addition to the geol...

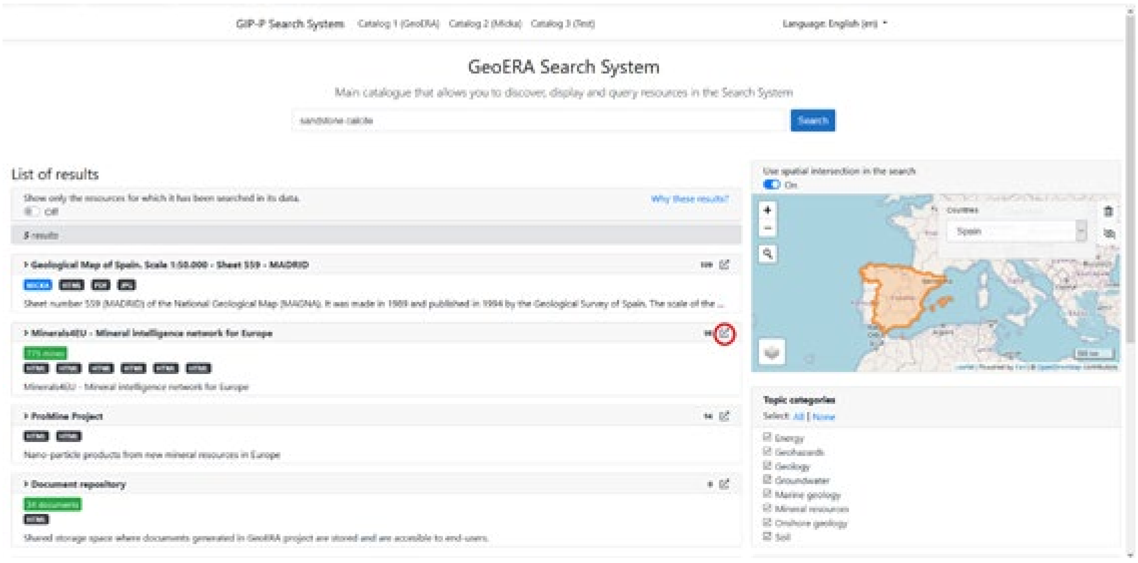
Use spatial intersection in the search
On
Countries
Spain

Topic categories
Select: All | None
☑ Energy
☑ Geohazards
☑ Geology
☑ Groundwater
☑ Marine geology
☑ Mineral resources
☑ Onshore geology
☑ Soil

Figure 11: Search system spatial selector

4.2.4.3 Stage 3

Type on the search text input box the string "sandstone calcite" and press the Search button.


GIP-P Search System Catalog 1 (GeoERA) Catalog 2 (Micka) Catalog 3 (Test) Language: English (en)

GeoERA Search System

Main catalogue that allows you to discover, display and query resources in the Search System

sandstone calcite

List of results

Show only the resources for which it has been searched in its data. Off

Why these results?

8 results

› Geological Map of Spain. Scale 1:50.000 - Sheet 559 - MADRID

Sheet number 559 (MADRID) of the National Geological Map (MAGNA). It was made in 1989 and published in 1994 by the Geological Survey of Spain. The scale of the ...

› Minerals4EU - Mineral intelligence network for Europe

775 mines

Minerals4EU - Mineral intelligence network for Europe

› ProMine Project

Nano-particle products from new mineral resources in Europe

› Document repository

34 documents

Shared storage space where documents generated in GeoERA project are stored and are accesible to end-users.

Use spatial intersection in the search: On

Countries: Spain

Topic categories

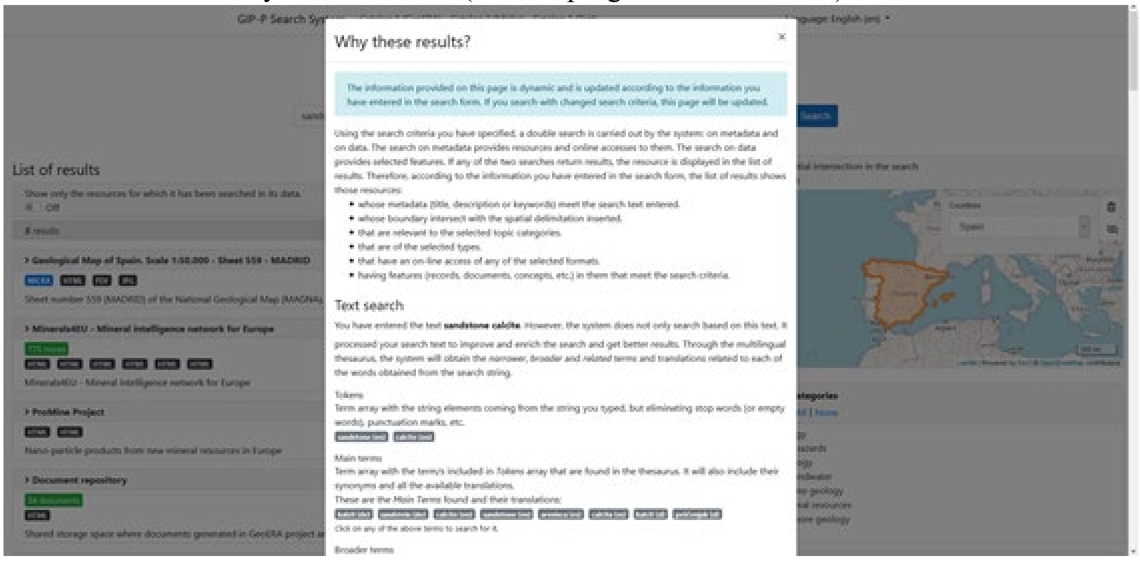
Select: All | None

☑ Energy
☑ Geohazards
☑ Geology
☑ Groundwater
☑ Marine geology
☑ Mineral resources
☑ Onshore geology
☑ Soil

Figure 12: The results

4.2.4.4 Stage 4

Click on the link “Why these results?” (at the top right of the result list).


Why these results?

The information provided on this page is dynamic and is updated according to the information you have entered in the search form. If you search with changed search criteria, this page will be updated.

Using the search criteria you have specified, a double search is carried out by the system: on metadata and on data. The search on metadata provides resources and online accesses to them. The search on data provides selected features. If any of the two searches return results, the resource is displayed in the list of results. Therefore, according to the information you have entered in the search form, the list of results shows those resources:

- whose metadata (title, description or keywords) meet the search text entered.
- whose boundary intersect with the spatial delimitation inserted.
- that are relevant to the selected topic categories.
- that are of the selected types.
- that have an on-line access of any of the selected formats.
- having features (records, documents, concepts, etc.) in them that meet the search criteria.

Text search

You have entered the text **sandstone calcite**. However, the system does not only search based on this text. It processed your search text to improve and enrich the search and get better results. Through the multilingual thesaurus, the system will obtain the narrower, broader and related terms and translations related to each of the words obtained from the search string.

Tokens

Term array with the string elements coming from the string you typed, but eliminating stop words (or empty words), punctuation marks, etc.

sandstone (en) calcite (en)

Main terms

Term array with the term/s included in Tokens array that are found in the thesaurus. It will also include their synonyms and all the available translations.

These are the Main Terms found and their translations:

[illegible]

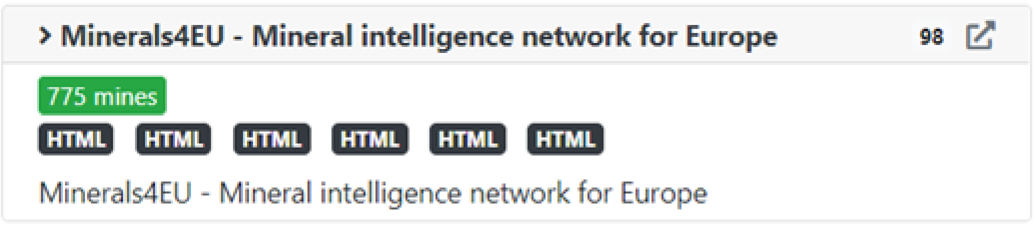
Click on any of the above terms to search for it.

Broader terms

Figure 13: The “Why these results?” popup

4.2.4.5 Stage 5

In order to close the “Why these results” modal page, click on the X (close icon). Then click on “775 mines”.


› Minerals4EU - Mineral intelligence network for Europe 98

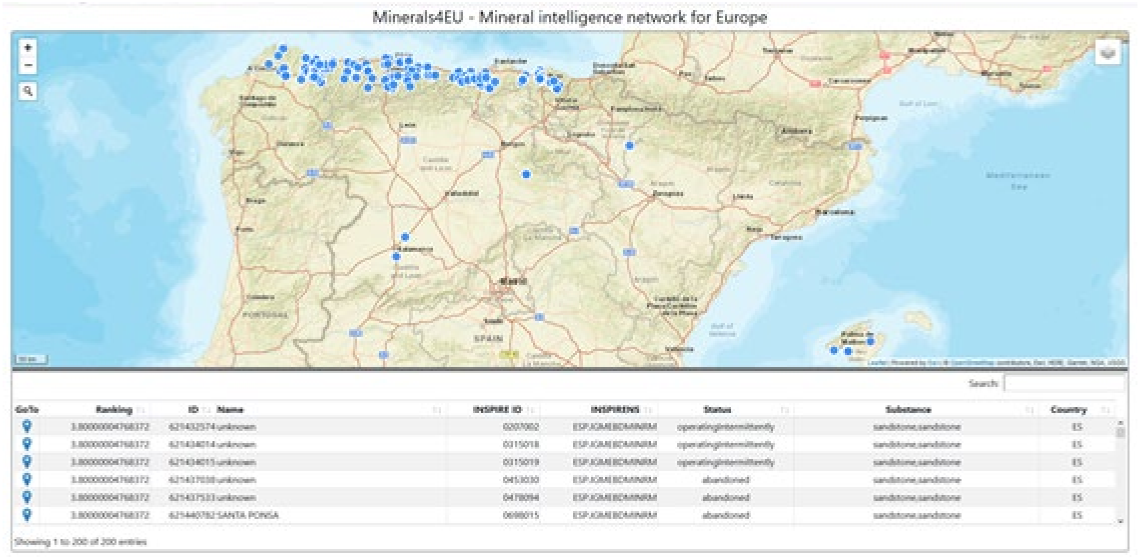
775 mines

HTML HTML HTML HTML HTML HTML

Minerals4EU - Mineral intelligence network for Europe

Figure 14

A new tab in the browser is opened showing the location of the selected mines and a table with their main attributes.


Minerals4EU - Mineral intelligence network for Europe

Search:

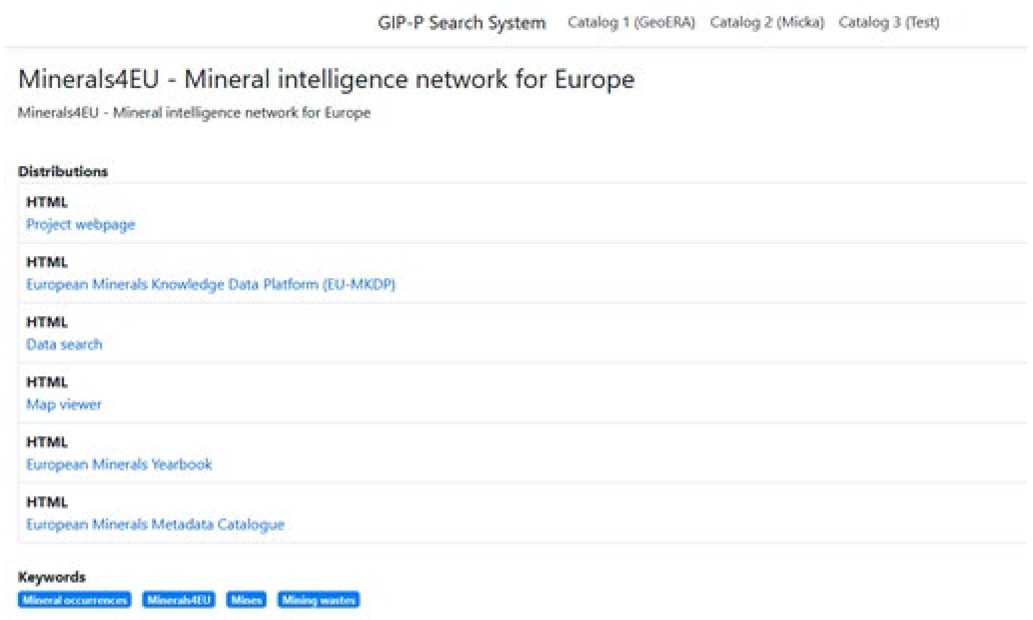
GoTo	Ranking	ID	Name	INSPIRE ID	INSPIRENS	Status	Substance	Country
📍	3.80000004768372	621432574	unknown	0207002	ESP.IGMEBDMINRM	operatingIntermittently	sandstone,sandstone	ES
📍	3.80000004768372	621434014	unknown	0315018	ESP.IGMEBDMINRM	operatingIntermittently	sandstone,sandstone	ES
📍	3.80000004768372	621434015	unknown	0315019	ESP.IGMEBDMINRM	operatingIntermittently	sandstone,sandstone	ES
📍	3.80000004768372	621437030	unknown	0453030	ESP.IGMEBDMINRM	abandoned	sandstone,sandstone	ES
📍	3.80000004768372	621437533	unknown	0478094	ESP.IGMEBDMINRM	abandoned	sandstone,sandstone	ES
📍	3.80000004768372	621440782	SANTA PONSA	0698015	ESP.IGMEBDMINRM	abandoned	sandstone,sandstone	ES

Showing 1 to 200 of 200 entries

Figure 15: The map with the locations of selected search results

4.2.4.6 Stage 6

Going back to the page shown in stage 3, press the icon ↗ (surrounded with the red circle in the image) to open the detailed information for the resource in a new window.



Minerals4EU - Mineral intelligence network for Europe

Minerals4EU - Mineral intelligence network for Europe

Distributions

HTML
Project webpage

HTML
European Minerals Knowledge Data Platform (EU-MKDP)

HTML
Data search

HTML
Map viewer

HTML
European Minerals Yearbook

HTML
European Minerals Metadata Catalogue

Keywords
Mineral occurrences | Minerals4EU | Mines | Mining wastes

Figure 16

4.2.5 Scalability issues

The Search System is based in microservices architecture, an architectural style that structures an application as a collection of services that are:

- Highly maintainable and testable.
- Loosely coupled.
- Independently deployable.
- Organized around business capabilities.
- Owned by a small team.

This offers good scalability, agility and reliability. In any case, the intention is to test this aspect and improve everything that is possible in the future, but the use of this architecture offers very good guarantees in this area.
More information about ASP.NET Core hosting in a web farm and its behaviour can be found in the following link, if needed:
https://docs.microsoft.com/en-us/aspnet/core/host-and-deploy/web-farm?view=aspnetcore-3.1

4.2.6 Users review

After the Search System's first demo, the URL to access the published draft version of the system was sent to all the IGME colleagues participating in any GeoERA projects. We warned them that it was a first version to which we had to add functionality, correct errors, add many more resources in the search, etc. However, as we are developing the system on prototypes, we thought it would be useful to get their feedback: usability, aspects that they think could be improved, non-intuitive things, etc.

So far, the only contribution proposed to add the possibility of collapsing the filters (by topic category, resource type and formats), in order to have better access to filters down in the page, especially when the lists are very long. This functionality has already been implemented.

4.2.7 Performance testing

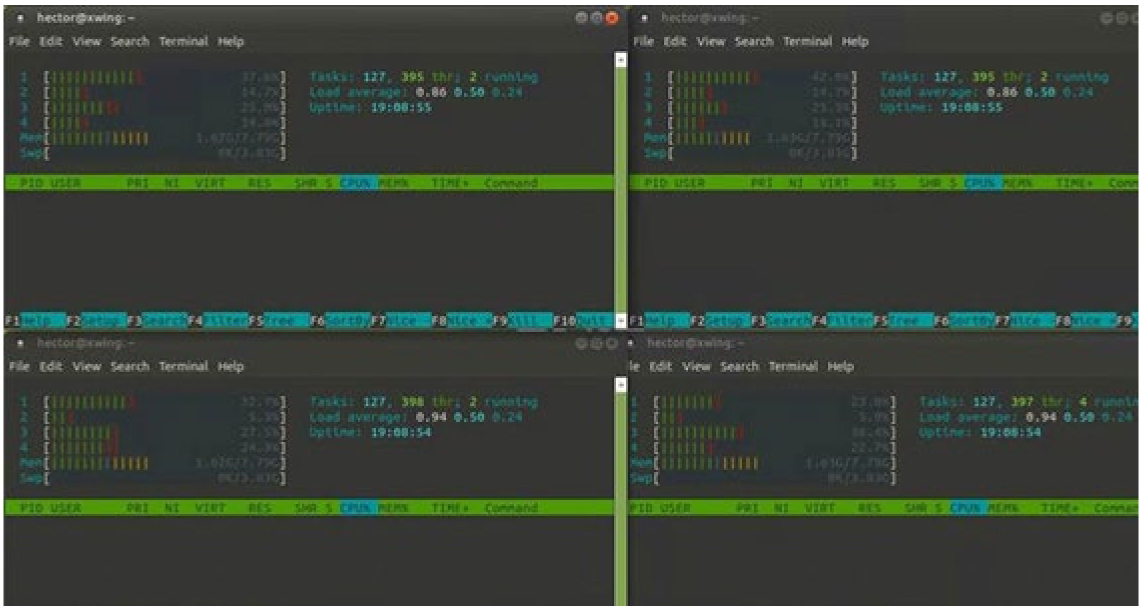
The automatic tests make:

- 2 calls to the Suggester API (Autocomplete Web API for Search System)
- 2 calls to the Spatial Selector API. One to get a list of territorial units and another one to get the geometry for a territorial until.
- 4 calls to the API that selects features inside resources.
- 1 call to the MVC Web application for the Search System.

These series of requests include the main types of calls that the Search System makes in a regular search process. This cycle is repeated multiple times and in parallel during the different tests performed.

4.2.7.1 Basic stress test

The system is exposed to a large number of looping (20) multithread (2) requests.



```
hector@xwing: ~
File  Edit  View  Search  Terminal  Help

  1  [|||||||||||||                37.8%]   Tasks: 127, 395 thr; 2 running
  2  [||||||                       14.7%]   Load average: 0.86 0.50 0.24
  3  [|||||||||                    25.0%]   Uptime: 19:08:55
  4  [|||||                        14.3%]
  Mem[||||||||||||||||      1.67G/7.79G]
  Swp[                         0K/3.83G]

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command

F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

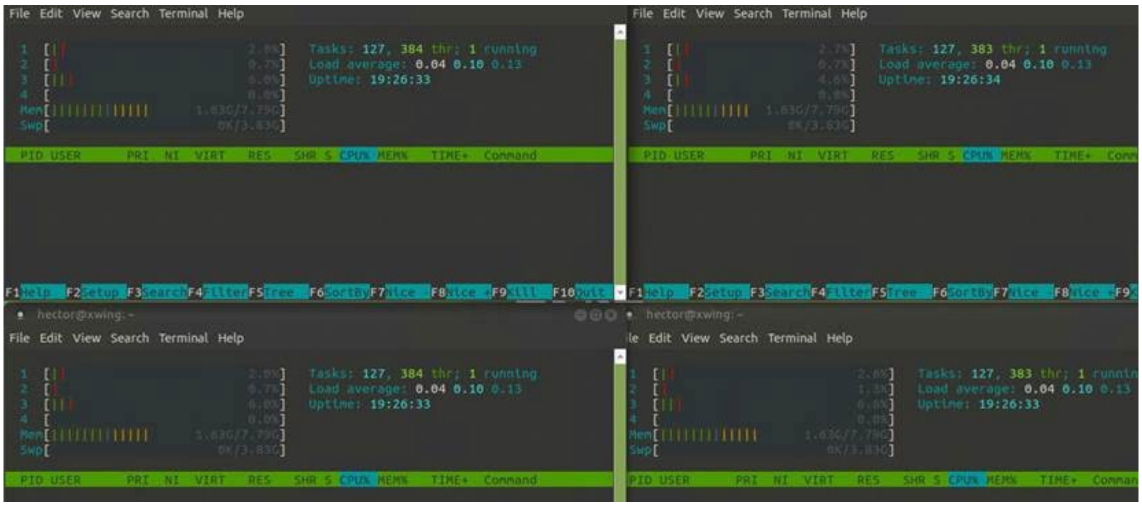
```
hector@xwing: ~
File  Edit  View  Search  Terminal  Help

  1  [|||||||||||||                42.0%]   Tasks: 127, 395 thr; 2 running
  2  [||||||                       14.7%]   Load average: 0.86 0.50 0.24
  3  [||||||||                     25.3%]   Uptime: 19:08:55
  4  [|||||                        16.1%]
  Mem[||||||||||||||        1.63G/7.79G]
  Swp[                         0K/3.83G]

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Comm

F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9
```

```
hector@xwing: ~
File  Edit  View  Search  Terminal  Help

  1  [|||||||||||                  32.7%]   Tasks: 127, 398 thr; 2 running
  2  [|||                           5.3%]   Load average: 0.94 0.50 0.24
  3  [|||||||||                    27.5%]   Uptime: 19:08:54
  4  [||||||||                     24.9%]
  Mem[||||||||||||||||      1.62G/7.79G]
  Swp[                         0K/3.83G]

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
```

```
hector@xwing: ~
le  Edit  View  Search  Terminal  Help

1  [||||||||                      23.0%]   Tasks: 127, 397 thr; 4 runnin
2  [|||                            5.9%]   Load average: 0.94 0.50 0.24
3  [|||||||||||||                 48.4%]   Uptime: 19:08:54
4  [|||||||                       22.7%]
Mem[||||||||||||||||        1.63G/7.79G]
Swp[                           0K/3.83G]

PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Comman
```

Figure 17: The consumption of memory does not exceed 1.7 GB.

4.2.7.2 System in standby



```
File  Edit  View  Search  Terminal  Help

  1  [||                            2.6%]   Tasks: 127, 384 thr; 1 running
  2  [|                             0.7%]   Load average: 0.04 0.10 0.13
  3  [|||                           6.0%]   Uptime: 19:26:33
  4  [                              0.0%]
  Mem[||||||||||||||||      1.63G/7.79G]
  Swp[                         0K/3.83G]

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command

F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9Kill  F10Quit
```

```
File  Edit  View  Search  Terminal  Help

  1  [||                            2.7%]   Tasks: 127, 383 thr; 1 running
  2  [|                             0.7%]   Load average: 0.04 0.10 0.13
  3  [||                            4.6%]   Uptime: 19:26:34
  4  [                              0.0%]
  Mem[||||||||||||||        1.63G/7.79G]
  Swp[                         0K/3.83G]

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Comm

F1Help  F2Setup F3SearchF4FilterF5Tree  F6SortByF7Nice -F8Nice +F9
```

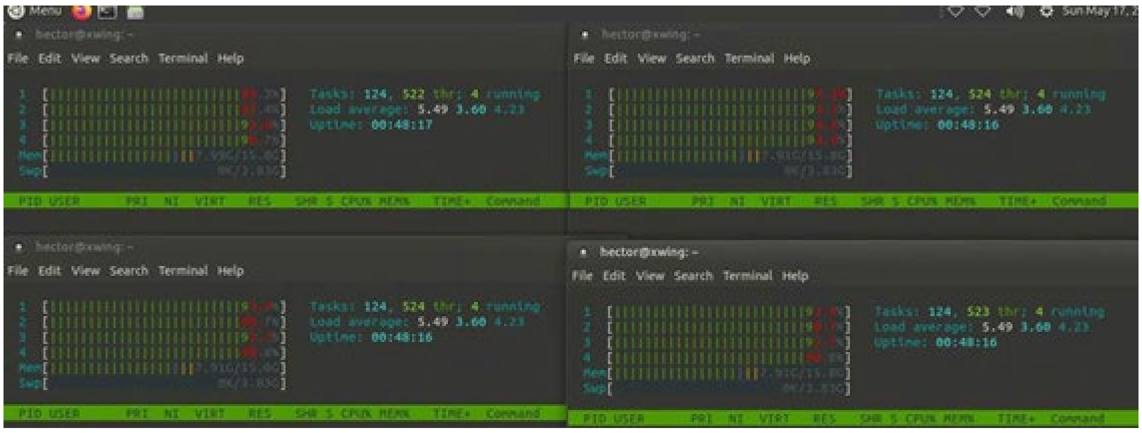
```
hector@xwing: ~
File  Edit  View  Search  Terminal  Help

  1  [||                            2.6%]   Tasks: 127, 384 thr; 1 running
  2  [|                             0.7%]   Load average: 0.04 0.10 0.13
  3  [|||                           6.0%]   Uptime: 19:26:33
  4  [                              0.0%]
  Mem[||||||||||||||||      1.63G/7.79G]
  Swp[                         0K/3.83G]

  PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
```

```
hector@xwing: ~
le  Edit  View  Search  Terminal  Help

1  [||                             2.6%]   Tasks: 127, 383 thr; 1 runnin
2  [|                              1.3%]   Load average: 0.04 0.10 0.13
3  [|||                            6.0%]   Uptime: 19:26:33
4  [                               0.0%]
Mem[||||||||||||||||        1.63G/7.79G]
Swp[                           0K/3.83G]

PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Comman
```

Figure 18: On standby, the system has an approximate consumption of 1 GB. This value is similar to the one used for the stress test; it seems to leave a similar amount of memory reserved for the test.

4.2.7.3 Advanced stress test

The system is subjected to a large number of looping (500) multithread (10) requests from two computers simultaneously. In total, 20 threads making 500 full calls each.



```
hector@xwing: ~
Tasks: 124, 522 thr; 4 running
Load average: 5.49 3.60 4.23
Uptime: 00:48:17
PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
```

```
hector@xwing: ~
Tasks: 124, 524 thr; 4 running
Load average: 5.49 3.60 4.23
Uptime: 00:48:16
PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
```

```
hector@xwing: ~
Tasks: 124, 524 thr; 4 running
Load average: 5.49 3.60 4.23
Uptime: 00:48:16
PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
```

```
hector@xwing: ~
Tasks: 124, 523 thr; 4 running
Load average: 5.49 3.60 4.23
Uptime: 00:48:16
PID USER      PRI  NI  VIRT   RES   SHR S CPU% MEM%   TIME+  Command
```

Figure 19: The consumption of memory does not exceed the 8 GB.

4.2.8 Planned improvements

The main improvements to include in the system are:

- To keep advancing in the functionality based on the established requirements and the users feedback after the different versions that are published.
- Incorporation into the system all the resources available in Micka and its adaptation to the system.
- Incorporation of feature distributions that allow searching features inside dataset (records in databases, documents in the document repository, etc).
- Integration with the web GIS for the visualization of the selected features in feature distributions.

4.3 3D viewer

The current 3D viewer is more a proof of concept than real 3D viewer. It is hoped that this very simple viewer will be replaced with a viewer with more functions. This is however not clear at the time of writing.

The viewer runs in a browser and all visualization is done at the computer showing the model. Therefore visualizing only puts a load on the central platform when reading the geometries from the database. This is done through two sets of REST services. One that delivers the setup of the model and one that deliver the spatial data. The data are delivered on a binary point cloud format for most efficient data delivery. This binary data are decoded in the browser to a format easily accessed from three.js.
As the final viewer is not in place, testing is not an issue currently.

In order to retrieve a model you will need its ModelId. When you have this you can retrieve information about the model and it layers using this call (for the model with ModelId=4):

http://geusegdi01.geus.dk/meta3d/rpc/model_meta?modelid=4

Based on the output of this call you can retrieve the different geometries / points clouds using this call:

http://geusegdi01.geus.dk/geom3d/data/nodes/68

For tin surfaces you will need two point clouds, one to get the vertices and one to get the triangles.

4.4 EGDI document repository search thematic application

4.4.1 Introduction

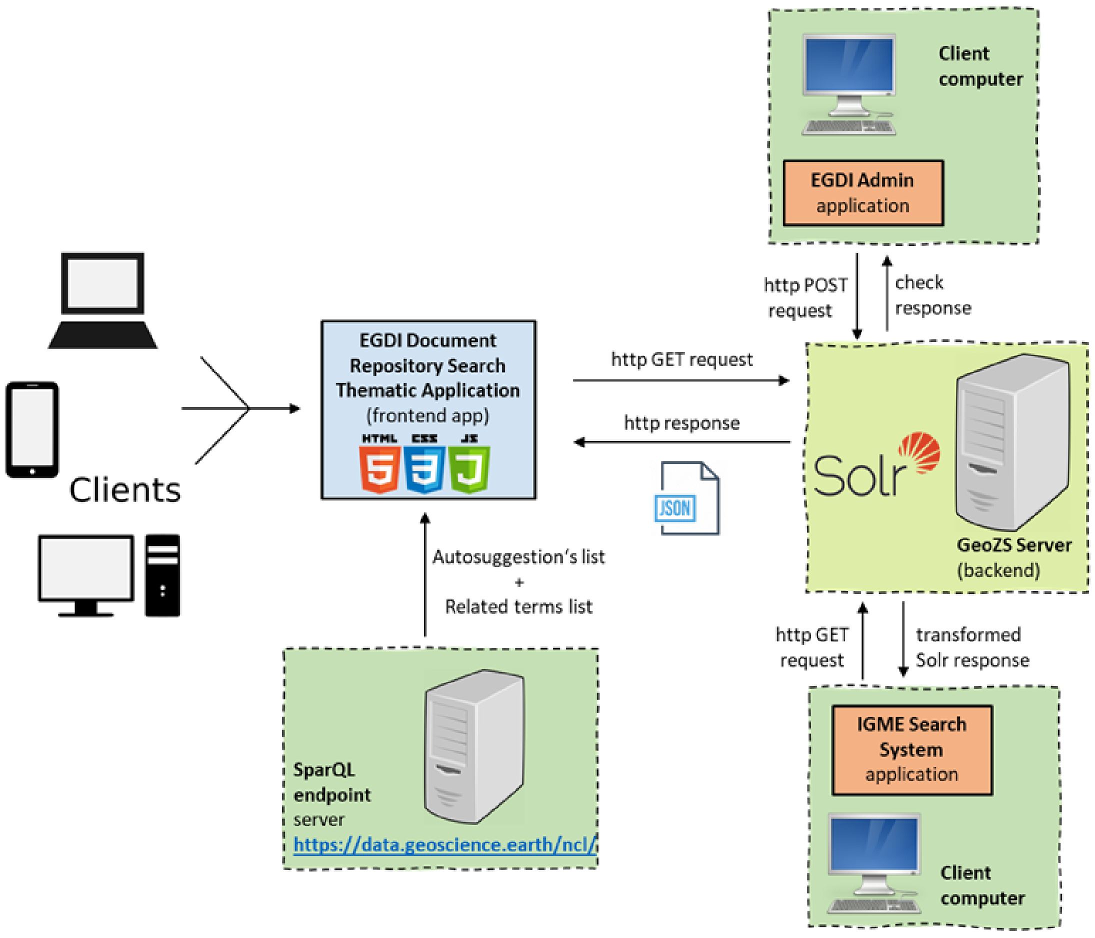
Several projects have requested the possibility to upload different types of unstructured data which can also contain different types of metadata. The GIP-P has thus decided to develop a simple document repository capable of storing PDF documents, pictures and tabular data. For documents, for which the projects have no ownership, it will be possible to register them through a DOI, so that they can be accessed through the portal. Documents with DOI links will not be stored in the repository. Therefore, they will be searchable through their metadata, but not through their content. All uploaded documents will be available via permanent links from the EGDI platform.
When a user from a project uploads a (main) document (and optionally, attachments to that document) into the EGDI document repository, the document and all attachments are stored on a filesystem at GEUS, metadata of the main document gets inserted into the repository database and are at the same time sent to Solr index at GeoZS so that they can be searched upon.
The 'EGDI document repository search' thematic application is the single-entry point through which the user interacts with the repository search system and makes his search(es). It is the application that runs in the users' browser, and which makes specific calls to different backend services to make the users search possible. It enables users to perform a detailed (searching through different metadata fields of a document including searching through the document content), thematic (search by related keywords) and ranked (results get evaluated based on the evaluation criteria) search through documents that are uploaded into the document repository through the EGDI admin portal.

4.4.2 Architecture

The EGDI Document Repository Search Thematic Application is currently available at the URL:https://egdi-search.geo-zs.si but the address may change before the final release (proposed https://search.europe-goelogy.eu).

It is the frontend application that runs in a user's browser and serves as a user interface to search through data collections that contain documents that are indexed into Solr using POST request via the EGDI Admin application.

Indexing the documents to Solr will be limited to EGDI Admin server only.



Client computer
EGDI Admin application
http POST request
check response
Clients
EGDI Document Repository Search Thematic Application (frontend app)
HTML CSS JS
http GET request
http response
JSON
Solr
GeoZS Server (backend)
Autosuggestion's list + Related terms list
http GET request
transformed Solr response
SparQL endpoint server
https://data.geoscience.earth/ncl/
IGME Search System application
Client computer

Figure 20: EGDI repository search thematic application architecture

Detailed description:

- The user visits the https://egdi-search.geo-zs.si page in his browser.
- The application begins to load, sending several HTTP GET requests to:
  - SparQL endpoint, to receive the data for the autosuggested keywords list
  - Solr, to receive the documents from different Solr cores and their corresponding metadata
- The application loads and enables the input box for the user to interact with.
- The user sets his search options and starts his search (detailed usage in chapter 2.7 - User/test scenario (Basic usage & User interface)).
- Based on the user's search options, searched keyword(s) and type of search (basic, semantic or advanced) the application constructs the query and makes the HTTP GET request to the Solr search engine.
- Solr browses through its index files to find the documents that match the searching criteria and responds with a JSON document.
- Application parses this JSON file and filters and renders its content to the user's browser for the user to browse through.

EGDI Repository Search platform (frontend application) should work in all the latest major browser versions:

- Chrome
- Firefox
- Opera
- Edge
- Safari (not tested yet)

At the time of writing version 1.3.2. (with current functionalities) also works in the legacy browser Internet Explorer 11.

4.4.2.1 Software

Module		**Type**	**Version**
HTML CSS JS		*Web technologies used for frontend of EGDI Document Repository Search Thematic Application*	
Apache Web Server or Microsoft IIS		*Web Server*	
Solr		*Search platform*	8.4.1
PostgreSQL		Database Manager	PostgreSQL 10.10
SPARQL		*Query Language for RDF (for fetching related terms)*	
Java OOP		*programming language; Solr requirement*	OpenJDK Runtime Environment 1.8.0

Windows Server or		Operating System	Independent

4.4.2.2 CPU and RAM

Virtual machine	CPUs	Memory	Disk space & technology
Virtual machine (for testing purpose)	Two virtual Intel Xeon E5-2650 v4 @2.2GHz CPU cores	4 GB RAM	64 GB (system) + 50 GB (other) NTFS

For production purposes we plan to install Solr on a machine with more resources, which includes more RAM, CPU power and faster disk.

4.4.2.3 Software disk space usage

The application is still in the development phase with only a few examples in Solr index, so the disk space usage will grow for Solr index files with an increasing number of documents, pictures and data files in the EGDI document repository.

Software element	Space disk (Windows)	Storage technology (Windows)
Frontend app (on web server)	1.21 MB	Hyper-v failover cluster with cluster shared volume (CSV) on HP MSA2052 and HP EVA6400 over FC. CSV utilizes the Windows NTFS file system on basic disks, leveraging GUID Partition Table (GPT) format disks.
Solr application	204 MB	Hyper-v failover cluster with cluster shared volume (CSV) on HP MSA2052 and HP EVA6400 over FC. CSV utilizes the Windows NTFS file system on basic disks, leveraging GUID Partition Table (GPT) format disks.
egdi-documents (solr index)	5 MB*	Hyper-v failover cluster with cluster shared volume (CSV) on HP MSA2052 and HP EVA6400 over FC. CSV utilizes the Windows NTFS file system on basic disks, leveraging GUID Partition Table (GPT) format disks.
egdi-images (solr index)	1 MB*	Hyper-v failover cluster with cluster shared volume (CSV) on HP MSA2052 and HP EVA6400 over FC. CSV utilizes the Windows NTFS file system on basic disks, leveraging GUID Partition Table (GPT) format disks.
egdi-data (solr index)	1 MB*	Hyper-v failover cluster with cluster shared volume (CSV) on HP MSA2052 and HP EVA6400 over FC. CSV utilizes the Windows NTFS file system on basic disks, leveraging GUID Partition Table (GPT) format disks.

4.4.3 Metrics

4.4.3.1 Number of expected users

We know the users will be from all of Europe, but we are unable to estimate the number of concurrent users, neither do we know how heavy the user's tasks will be. The real case scenario is so far unknown, but we develop the application with scalability in mind.

4.4.3.2 Number of requests (HTTP GET/POST/PUT…)

To reduce the number of requests needed to load all the data and to limit the transferred data size, only a partial number of results will be returned to the user when he starts the search. Additional requests will be sent only when the user wants to view the rest of the results by adding pagination in combination with GET requests.

4.4.4 Envisioned scenario for performance testing

Here we describe the simplest scenario of usage, because we anticipate that it will be the most used.

Empty browser cache

4.4.4.1 Stage 1

Go to https://www.geo-zs.si/db/egdi-search/

The site loads:

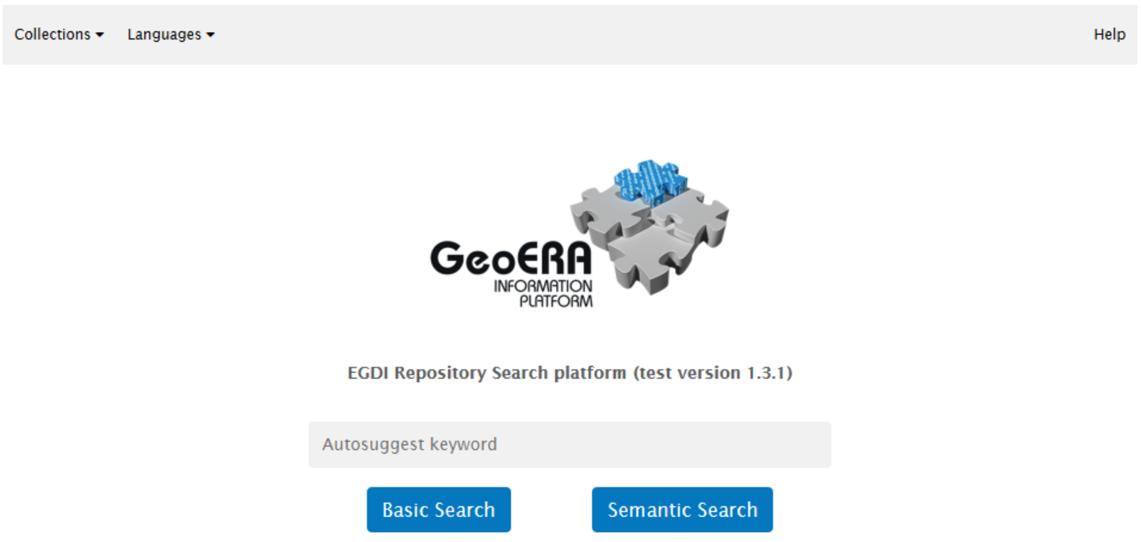
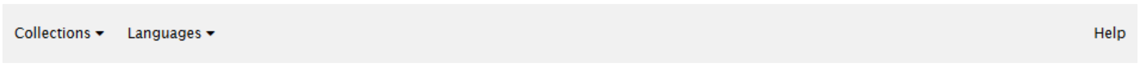

Collections
Languages
Help
GeoERA INFORMATION PLATFORM
EGDI Repository Search platform (test version 1.3.1)
Autosuggest keyword
Basic Search
Semantic Search

Figure 21: EGDI Repository search platform home page

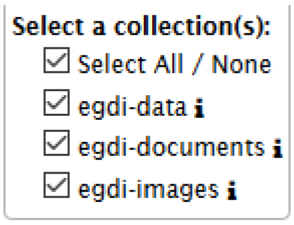
4.4.4.2 Stage 2

At the top of the loaded site lies the settings menu. Check its Collections and Languages options.



Collections ▾ Languages ▾ Help

If all collections are not already selected, check Select all option in the Collections menu:

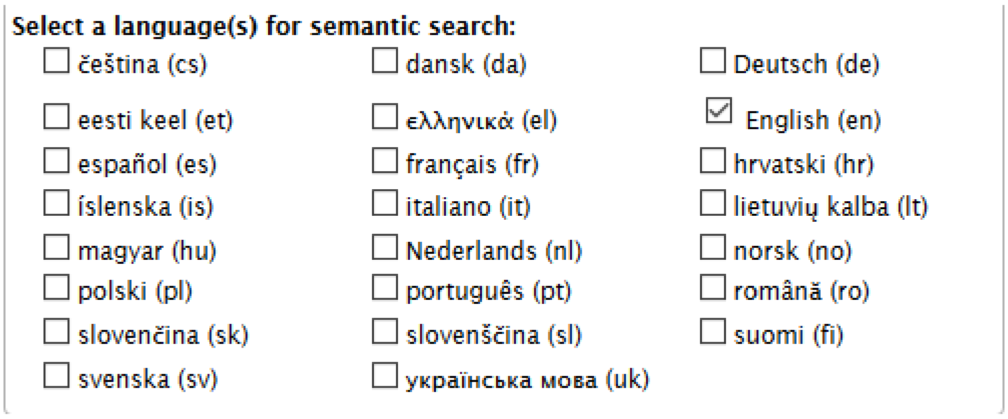


Select a collection(s):

- [✓ Select All / None]
- [✓ egdi-data] ℹ
- [✓ egdi-documents] ℹ
- [✓ egdi-images] ℹ

Figure 22: Settings menu

Select English (if it is not already selected) in the Languages menu as the language for semantic search.



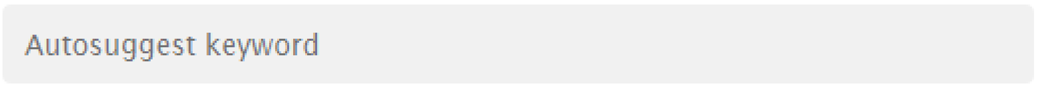
Select a language(s) for semantic search:

☐ čeština (cs)	☐ dansk (da)	☐ Deutsch (de)
☐ eesti keel (et)	☐ ελληνικά (el)	[✓ English (en)]
☐ español (es)	☐ français (fr)	☐ hrvatski (hr)
☐ íslenska (is)	☐ italiano (it)	☐ lietuvių kalba (lt)
☐ magyar (hu)	☐ Nederlands (nl)	☐ norsk (no)
☐ polski (pl)	☐ português (pt)	☐ română (ro)
☐ slovenčina (sk)	☐ slovenščina (sl)	☐ suomi (fi)
☐ svenska (sv)	☐ українська мова (uk)	

Figure 23: Language menu for semantic search

4.4.4.3 Stage 3

Set focus to the search field.

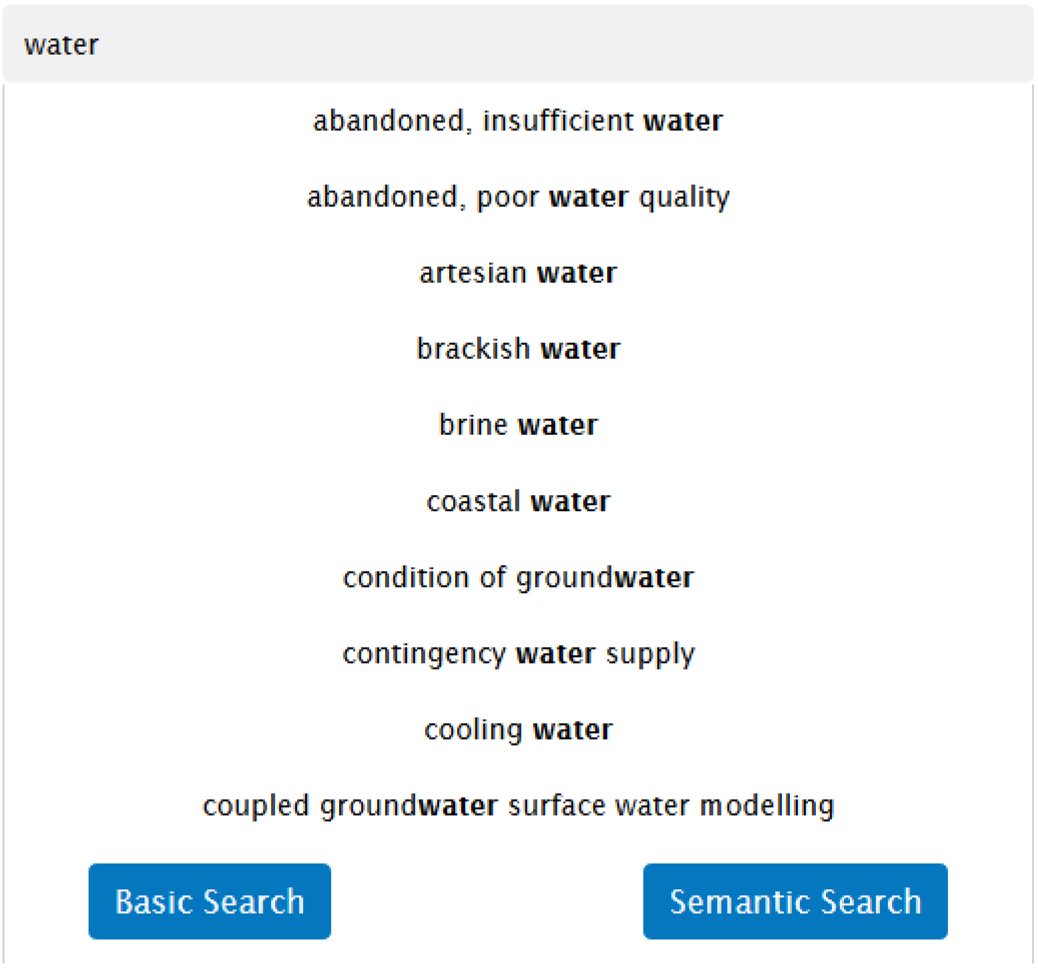


Autosuggest keyword

Figure 24: The search field

4.4.4.4 Stage 4

Start typing "water". As you type, a list of suggestions pops up. This way you can select the searched keyword, e.g.: "cooling water" with a mouse click in the suggestion's list or finish by typing the searched word manually.


water

abandoned, insufficient **water**

abandoned, poor **water** quality

artesian **water**

brackish **water**

brine **water**

coastal **water**

condition of ground**water**

contingency **water** supply

cooling **water**

coupled ground**water** surface water modelling

Figure 25:Autosuggester

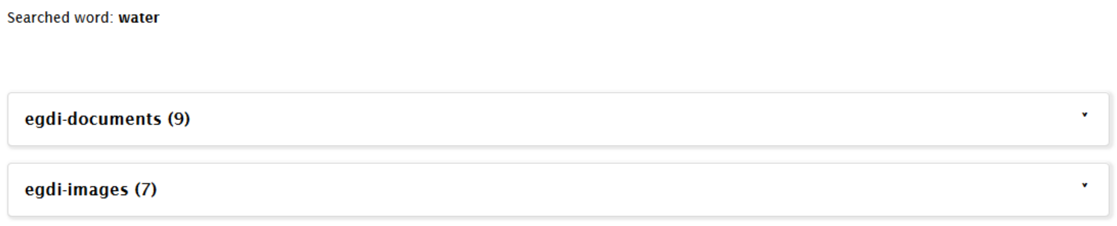
4.4.4.5 Stage 5

Now click on the Basic search button to perform the basic search or Semantic search button to perform the semantic search. (This stage will be changed when the type of the search is determined in the settings options)

4.4.4.6 Stage 6

Based on the type of the search previously selected the results are shown.

For Basic search:

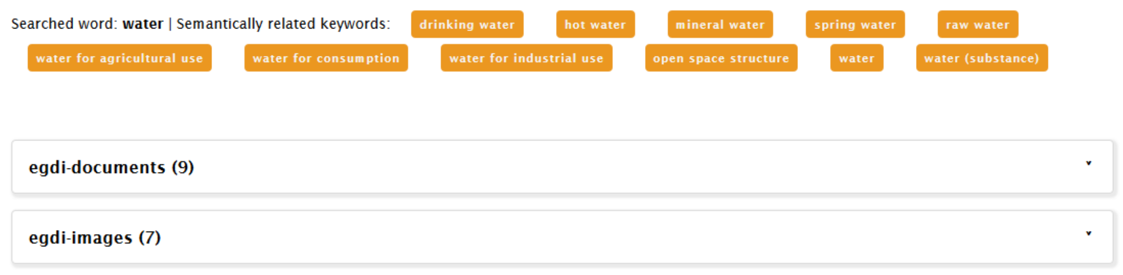

Searched word: **water**

egdi-documents (9)

egdi-images (7)

Figure 26: Basic search

For Semantic search:


Searched word: **water** | Semantically related keywords: drinking water hot water mineral water spring water raw water water for agricultural use water for consumption water for industrial use open space structure water water (substance)

egdi-documents (9)

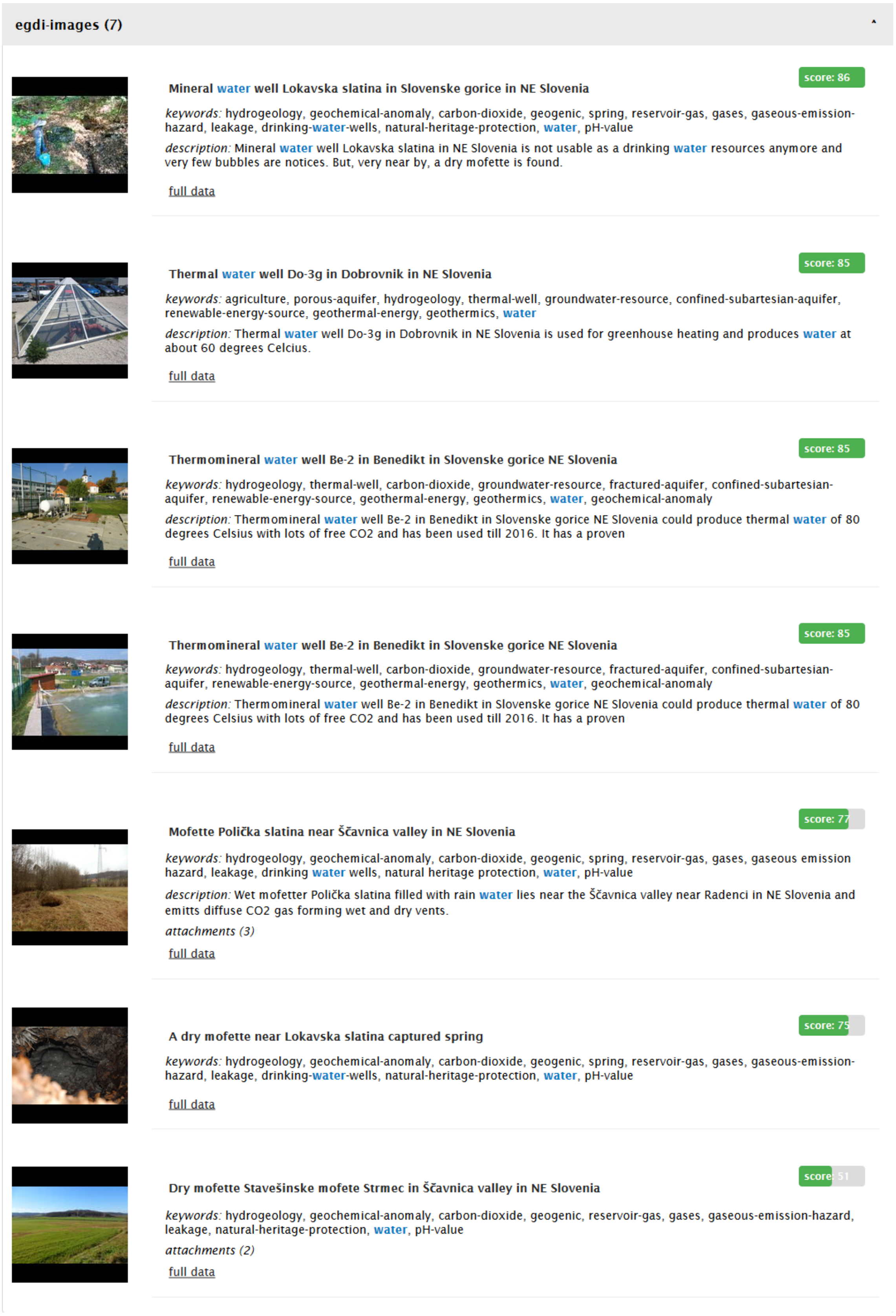
egdi-images (7)

Figure 27: Thematic search

Results show the collections which returned some documents based on the search criteria. In this example 0 (zero) documents were found in the egdi-data collection, so there is no tab for this collection, just for the egdi-documents and egdi-images.

If you click on the icon ˅ in the rightmost part of the collection's tab, the results for the specific collection are expanded.

For Basic search:



egdi-images (7)

score: 86

Mineral water well Lokavska slatina in Slovenske gorice in NE Slovenia

keywords: hydrogeology, geochemical-anomaly, carbon-dioxide, geogenic, spring, reservoir-gas, gases, gaseous-emission-hazard, leakage, drinking-water-wells, natural-heritage-protection, water, pH-value

description: Mineral water well Lokavska slatina in NE Slovenia is not usable as a drinking water resources anymore and very few bubbles are notices. But, very near by, a dry mofette is found.

full data

score: 85

Thermal water well Do-3g in Dobrovnik in NE Slovenia

keywords: agriculture, porous-aquifer, hydrogeology, thermal-well, groundwater-resource, confined-subartesian-aquifer, renewable-energy-source, geothermal-energy, geothermics, water

description: Thermal water well Do-3g in Dobrovnik in NE Slovenia is used for greenhouse heating and produces water at about 60 degrees Celcius.

full data

score: 85

Thermomineral water well Be-2 in Benedikt in Slovenske gorice NE Slovenia

keywords: hydrogeology, thermal-well, carbon-dioxide, groundwater-resource, fractured-aquifer, confined-subartesian-aquifer, renewable-energy-source, geothermal-energy, geothermics, water, geochemical-anomaly

description: Thermomineral water well Be-2 in Benedikt in Slovenske gorice NE Slovenia could produce thermal water of 80 degrees Celsius with lots of free CO2 and has been used till 2016. It has a proven

full data

score: 85

Thermomineral water well Be-2 in Benedikt in Slovenske gorice NE Slovenia

keywords: hydrogeology, thermal-well, carbon-dioxide, groundwater-resource, fractured-aquifer, confined-subartesian-aquifer, renewable-energy-source, geothermal-energy, geothermics, water, geochemical-anomaly

description: Thermomineral water well Be-2 in Benedikt in Slovenske gorice NE Slovenia could produce thermal water of 80 degrees Celsius with lots of free CO2 and has been used till 2016. It has a proven

full data

score: 77

Mofette Polička slatina near Ščavnica valley in NE Slovenia

keywords: hydrogeology, geochemical-anomaly, carbon-dioxide, geogenic, spring, reservoir-gas, gases, gaseous emission hazard, leakage, drinking water wells, natural heritage protection, water, pH-value

description: Wet mofetter Polička slatina filled with rain water lies near the Ščavnica valley near Radenci in NE Slovenia and emitts diffuse CO2 gas forming wet and dry vents.

attachments (3)

full data

score: 75

A dry mofette near Lokavska slatina captured spring

keywords: hydrogeology, geochemical-anomaly, carbon-dioxide, geogenic, spring, reservoir-gas, gases, gaseous-emission-hazard, leakage, drinking-water-wells, natural-heritage-protection, water, pH-value

full data

score: 51

Dry mofette Stavešinske mofete Strmec in Ščavnica valley in NE Slovenia

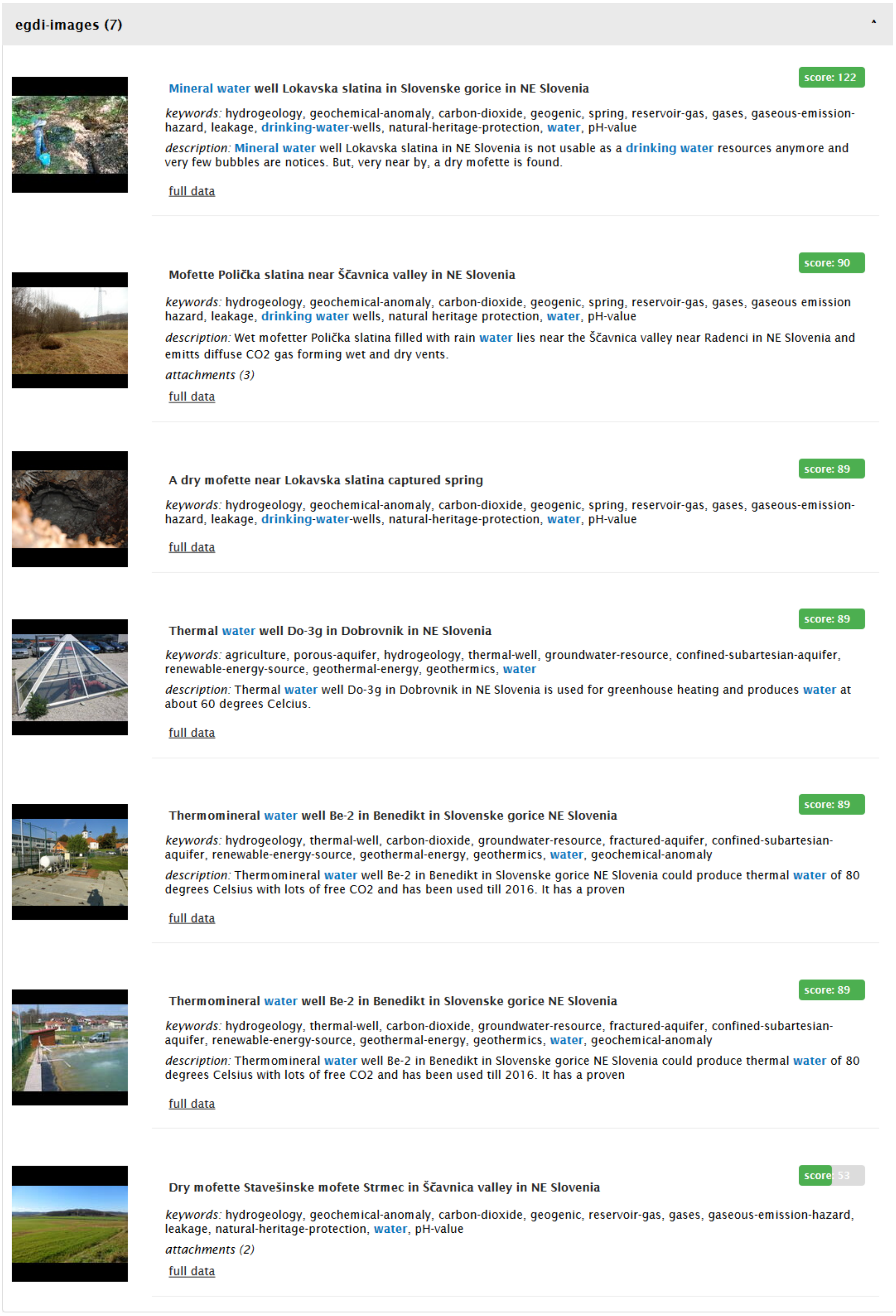
keywords: hydrogeology, geochemical-anomaly, carbon-dioxide, geogenic, reservoir-gas, gases, gaseous-emission-hazard, leakage, natural-heritage-protection, water, pH-value

attachments (2)

full data

Figure 28: Results after egdi-images expand

For Semantic search:



egdi-images (7)

score: 122

Mineral water well Lokavska slatina in Slovenske gorice in NE Slovenia

keywords: hydrogeology, geochemical-anomaly, carbon-dioxide, geogenic, spring, reservoir-gas, gases, gaseous-emission-hazard, leakage, drinking-water-wells, natural-heritage-protection, water, pH-value

description: Mineral water well Lokavska slatina in NE Slovenia is not usable as a drinking water resources anymore and very few bubbles are notices. But, very near by, a dry mofette is found.

full data

score: 90

Mofette Polička slatina near Ščavnica valley in NE Slovenia

keywords: hydrogeology, geochemical-anomaly, carbon-dioxide, geogenic, spring, reservoir-gas, gases, gaseous emission hazard, leakage, drinking water wells, natural heritage protection, water, pH-value

description: Wet mofetter Polička slatina filled with rain water lies near the Ščavnica valley near Radenci in NE Slovenia and emitts diffuse CO2 gas forming wet and dry vents.

attachments (3)

full data

score: 89

A dry mofette near Lokavska slatina captured spring

keywords: hydrogeology, geochemical-anomaly, carbon-dioxide, geogenic, spring, reservoir-gas, gases, gaseous-emission-hazard, leakage, drinking-water-wells, natural-heritage-protection, water, pH-value

full data

score: 89

Thermal water well Do-3g in Dobrovnik in NE Slovenia

keywords: agriculture, porous-aquifer, hydrogeology, thermal-well, groundwater-resource, confined-subartesian-aquifer, renewable-energy-source, geothermal-energy, geothermics, water

description: Thermal water well Do-3g in Dobrovnik in NE Slovenia is used for greenhouse heating and produces water at about 60 degrees Celcius.

full data

score: 89

Thermomineral water well Be-2 in Benedikt in Slovenske gorice NE Slovenia

keywords: hydrogeology, thermal-well, carbon-dioxide, groundwater-resource, fractured-aquifer, confined-subartesian-aquifer, renewable-energy-source, geothermal-energy, geothermics, water, geochemical-anomaly

description: Thermomineral water well Be-2 in Benedikt in Slovenske gorice NE Slovenia could produce thermal water of 80 degrees Celsius with lots of free CO2 and has been used till 2016. It has a proven

full data

score: 89

Thermomineral water well Be-2 in Benedikt in Slovenske gorice NE Slovenia

keywords: hydrogeology, thermal-well, carbon-dioxide, groundwater-resource, fractured-aquifer, confined-subartesian-aquifer, renewable-energy-source, geothermal-energy, geothermics, water, geochemical-anomaly

description: Thermomineral water well Be-2 in Benedikt in Slovenske gorice NE Slovenia could produce thermal water of 80 degrees Celsius with lots of free CO2 and has been used till 2016. It has a proven

full data

score: 53

Dry mofette Stavešinske mofete Strmec in Ščavnica valley in NE Slovenia

keywords: hydrogeology, geochemical-anomaly, carbon-dioxide, geogenic, reservoir-gas, gases, gaseous-emission-hazard, leakage, natural-heritage-protection, water, pH-value

attachments (2)

full data

Figure 29: Results after egdi-images expand

You can see the highlighted (with blue) words for the semantic search also include the keywords the search system got from the list of the related keywords.

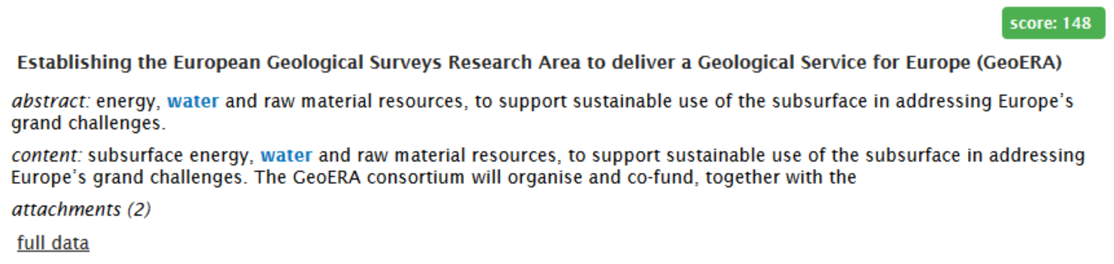
Documents inside a collection are sorted based on the score result. The score bar gives you information how a specific document is evaluated based on the number and weight of the found matching criteria for each document.

4.4.4.7 Stage 7

If the user clicks on the title of the document, e.g.: *Estimation of effective porosity in large-scale groundwater models by combining particle tracking, auto-calibration and 14C dating* the document's full article opens in a separate browser tab.

4.4.4.8 Stage 8

For a specific document, the user gets the information about which parts of the document contain the searched or related (for semantic search) keywords.


score: 148

Establishing the European Geological Surveys Research Area to deliver a Geological Service for Europe (GeoERA)

abstract: energy, water and raw material resources, to support sustainable use of the subsurface in addressing Europe's grand challenges.

content: subsurface energy, water and raw material resources, to support sustainable use of the subsurface in addressing Europe's grand challenges. The GeoERA consortium will organise and co-fund, together with the

attachments (2)

full data

Figure 30: Which parts (metadata) contain the searched od related keywords

4.4.4.9 Stage 9

If the user hovers over the attachments category, the list of the attachments related to the document is shown, e.g.:

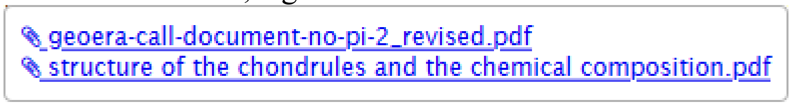

geoera-call-document-no-pi-2_revised.pdf
structure of the chondrules and the chemical composition.pdf

Figure 31: Attachment list

Any of these attachments are clickable and lead to the opening of new tabs of the browser with related data.

4.4.4.10 Stage 10

Attachments in the egdi-images collection are small thumbnails made from images. Clicking on each of them results in the opening of its full scale image.

4.4.4.11 Stage 11

Information about the document have the full data link at the document's bottommost part of its section. This is a link to full metadata information about this document in the collection that also opens in a new tab.

4.4.5 Scalability issues

4.4.5.1 General note

Generally, we can only say that frontend and backend are based on tools which have a great possibility for scalability.

4.4.5.2 Scalability on the frontend side

Simultaneous connections on an Apache server depend on the response time of each thread and the memory used by each thread. Since dynamic sites are powered by databases on the backend, it becomes very difficult to guess the amount of traffic a single Apache server can handle. Of course, we do not want that the total memory used by all the threads exceeds the system memory.

Scalability related to the frontend application will be implemented in a relation to the optimization and reduction of the GET requests that are fired based on some user interaction with the application's user interface.

When the search request is made, the asynchronous HTTP GET requests are made in parallel to all available Solr cores and when the frontend application gets results from the last Solr HTTP GET request, the data is merged into single JSON object. While a part of the application waits for the results from the requests to the Solr, the application itself stays responsive, because of the asynchronous nature of the HTTP GET requests.

4.4.5.3 Scalability on the backend side

Solr is a highly scalable search platform. It uses a denormalized document data model (each document is self-contained, independent of one another). Good schema design is an important factor to enhance the scalability of Solr.

Besides that, Solr offers:
1. Cache-management features,
2. Distribution of index: Divide the index (with many documents) into shards, each of which runs on a separate machine and perform distributed search,
3. Replication of index on multiple servers.

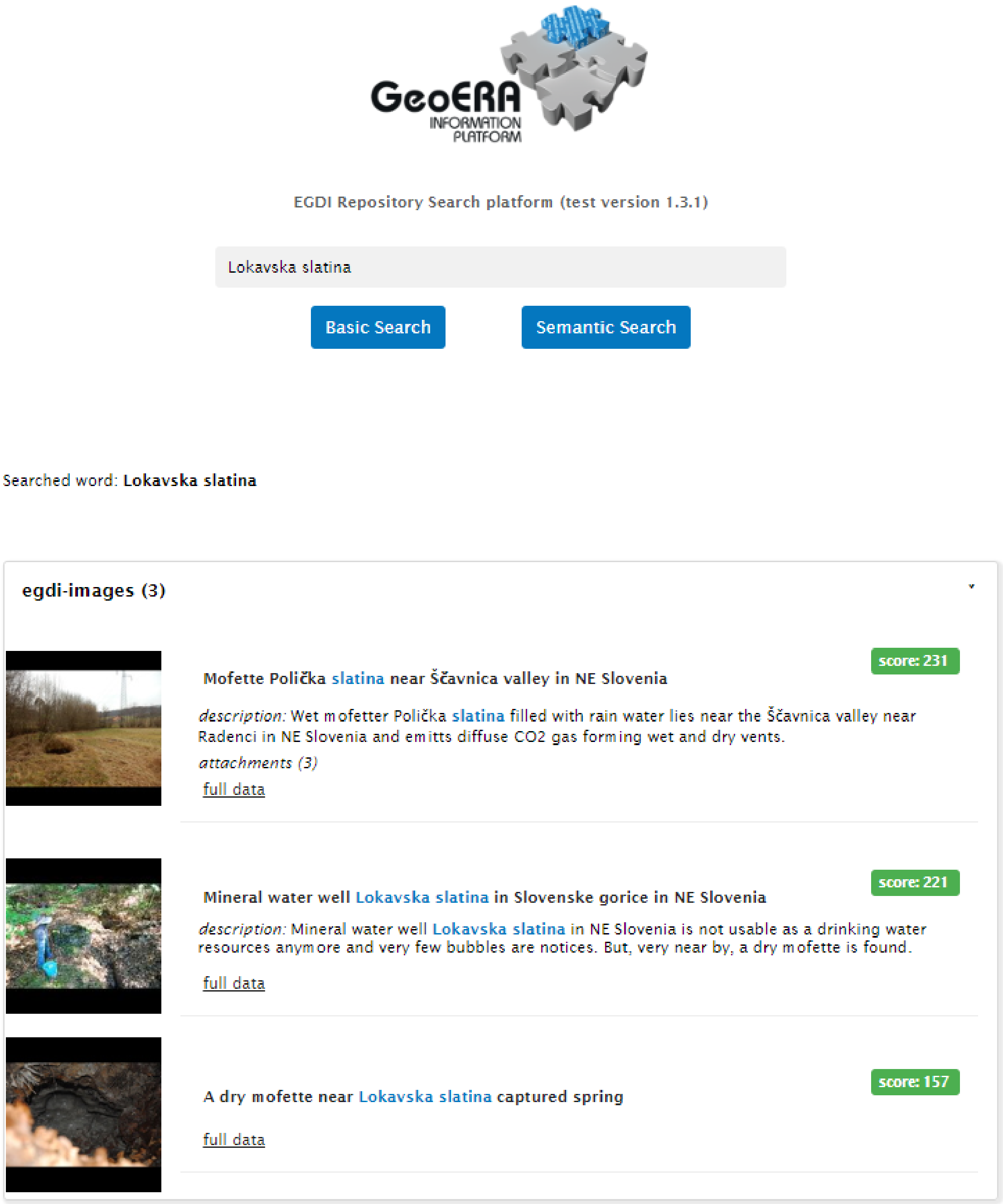
The SolrCloud (with an embedded ZooKeeper) can also be used for this purpose.

4.4.5.4 Testing

Test indexing (and searching) is performed after every change in configuration of a Solr core.

4.4.6 Users review

The first test of the 'EGDI Repository Search platform' was done from GIP-P project members, and first test from external user was provided by member of HIKE project but at that time only the searching through a pdf document was possible. The image search was implemented later, and the first demonstration/test was done by Geoconnect3D members on February 28th, and additional demonstration/test with improved application was provided on May 14th to the Geoconnect3D project leader and other project members.


EGDI Repository Search platform (test version 1.3.1)

Lokavska slatina

Basic Search | Semantic Search

Searched word: **Lokavska slatina**

egdi-images (3)

Mofette Polička slatina near Ščavnica valley in NE Slovenia — score: 231

description: Wet mofetter Polička slatina filled with rain water lies near the Ščavnica valley near Radenci in NE Slovenia and emitts diffuse CO2 gas forming wet and dry vents.

attachments (3)

full data

Mineral water well Lokavska slatina in Slovenske gorice in NE Slovenia — score: 221

description: Mineral water well Lokavska slatina in NE Slovenia is not usable as a drinking water resources anymore and very few bubbles are notices. But, very near by, a dry mofette is found.

full data

A dry mofette near Lokavska slatina captured spring — score: 157

full data

Figure 32: egdi-images search example

They asked for some mirror improvements that have been mostly implemented in the current 1.3.1 test version. Some proposals like changes in keywords entry, entering spatial search request data and nicer display of all document metadata we will try to implement in the future version of the app. Overall they are satisfied, and we received a good review from them.

4.4.7 Performance testing

So far there has been no performance testing. The application is in the middle of the development phase, so only basic tests as described in the previous paragraph were provided by some users.

4.4.8 Planned improvement

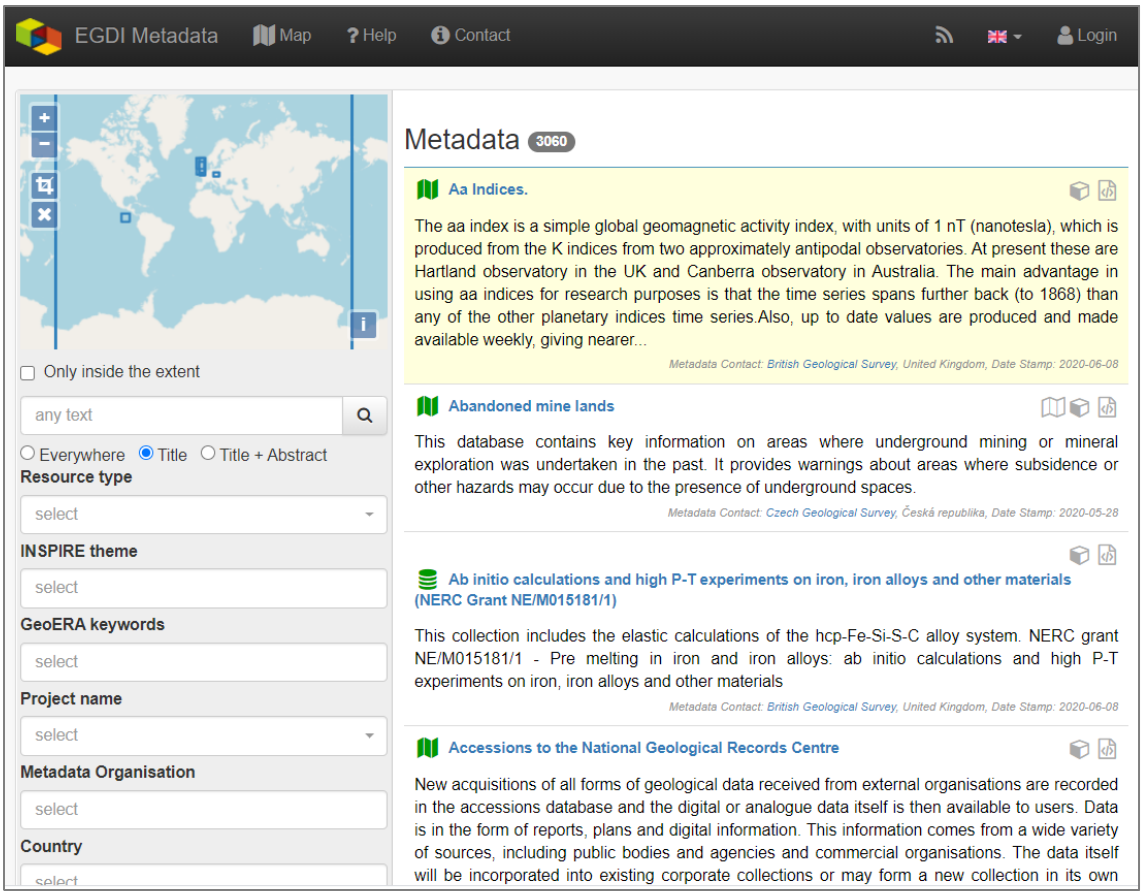
In the near future we plan some of the following improvements:

- Maximally reduce the number of requests needed to load all the data and to limit the transferred data size. Only a partial number of results will be returned to the user when he starts the search. Additional requests will be sent only when the user wants to view the rest of the results by adding pagination in combination with GET requests.
- Instead of using 'Basic Search' and 'Semantic Search' buttons use only 'Search' button and use radio button for selecting ''Basic Search' or 'Semantic Search' option.
- Add a map for selecting spatial search boundary, with additional radio buttons on/of for spatial search and contains/intersect radio buttons.
- Include MIcKA or Egdi-admin document upload modules for entering keywords. Be careful because additional possibility for advance search (:, OR, AND, branches).
- Replace current SparQL search database endpoint https://resource.geolba.ac.at/PoolParty/sparql/geoera_keyword (autosuggest & semantic) with current official version 2.0 with data.geoscience.earth URI https://data.geoscience.earth/ncl/ui/sparql-form
- Separate semantic words by languages in Related keywords section: 'Searched word: water | Semantically related keywords:'
- When using advance search like 'author:BAVEC' put only the author fields in results
- Include Broader/Narrower/Related Terms. Use the right service.
- Check the url limitation. The search url can be very long especially when users provide a semantic search with many languages selected. We need to examine if the browser, Solr or something else is limiting the available length of the searched url.
- Creating a site for bug and improvement reporting (at geusgitlab.dk) for external users.

4.5 MIcKA: EGDI metadata catalogue

4.5.1 Introduction

The EGDI Metadata Catalogue is the central access point to metadata concerning structured data on geo-energy, groundwater and raw materials themes provided by the geoscientific GeoERA projects. It provides tools for compilation of those metadata in a standardized format. In order to make the data discoverable in the most efficient way, the catalogue is fully compliant with international standards and supports the distributed system of metadata administration. In order to display a metadata record for which an online map service is available, the Metadata Catalogue is integrated into the EGDI Portal http://www.europe-geology.eu/. The catalogue enables systematic discovery, viewing and use of available geological data across Europe. The working version of the EGDI metadata catalogue is operational at https://egdi.geology.cz/ and on the project portal http://www.europe-geology.eu/metadata/.



Metadata 3060

Aa Indices.

The aa index is a simple global geomagnetic activity index, with units of 1 nT (nanotesla), which is produced from the K indices from two approximately antipodal observatories. At present these are Hartland observatory in the UK and Canberra observatory in Australia. The main advantage in using aa indices for research purposes is that the time series spans further back (to 1868) than any of the other planetary indices time series.Also, up to date values are produced and made available weekly, giving nearer...

Metadata Contact: British Geological Survey, United Kingdom, Date Stamp: 2020-06-08

Abandoned mine lands

This database contains key information on areas where underground mining or mineral exploration was undertaken in the past. It provides warnings about areas where subsidence or other hazards may occur due to the presence of underground spaces.

Metadata Contact: Czech Geological Survey, Česká republika, Date Stamp: 2020-05-28

Ab initio calculations and high P-T experiments on iron, iron alloys and other materials (NERC Grant NE/M015181/1)

This collection includes the elastic calculations of the hcp-Fe-Si-S-C alloy system. NERC grant NE/M015181/1 - Pre melting in iron and iron alloys: ab initio calculations and high P-T experiments on iron, iron alloys and other materials

Metadata Contact: British Geological Survey, United Kingdom, Date Stamp: 2020-06-08

Accessions to the National Geological Records Centre

New acquisitions of all forms of geological data received from external organisations are recorded in the accessions database and the digital or analogue data itself is then available to users. Data is in the form of reports, plans and digital information. This information comes from a wide variety of sources, including public bodies and agencies and commercial organisations. The data itself will be incorporated into existing corporate collections or may form a new collection in its own

Figure 33: The EGDI Metadata Catalogue

4.5.2 Architecture

4.5.2.1 Production environment

The system runs on the operating system SuSE Linux, on encrypted communication https://egdi.geology.cz/ during harvesting. After any major intervention on the server, a complete server backup is created on the virtualization platform level to enable swift recovery if needed. This backup is used as a clone of the server for testing of major application changes. Regular database increments and changes in the application code are backed up once a week. In addition to that, one backup copy for each month is stored for 12 months.

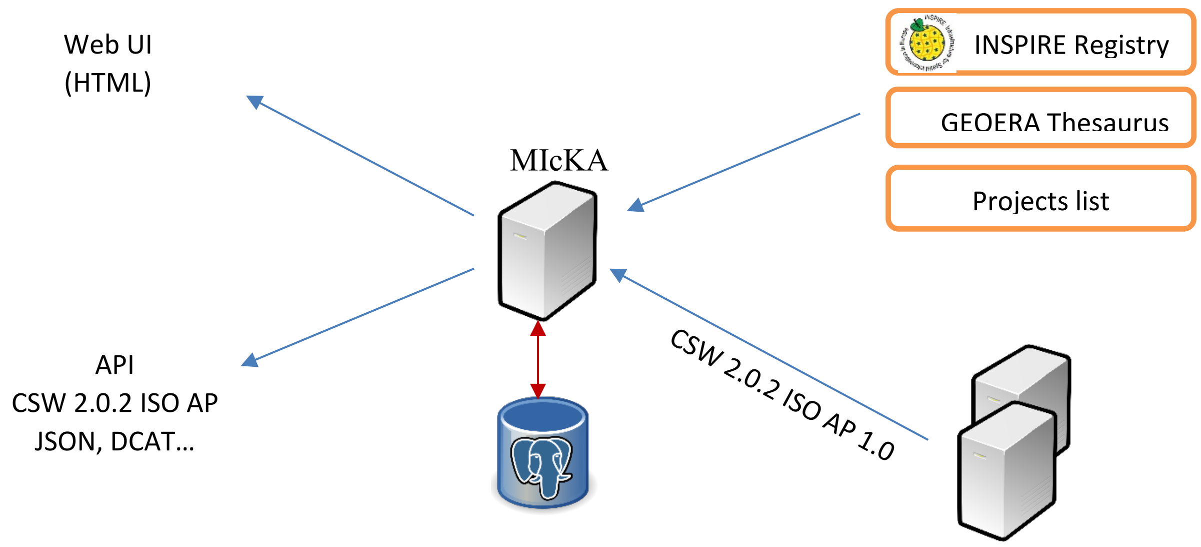


Figure 34: EGDI metadata catalogue (MIcKA) architecture

Internal schema of EGDI metadata catalogue (MIcKA)

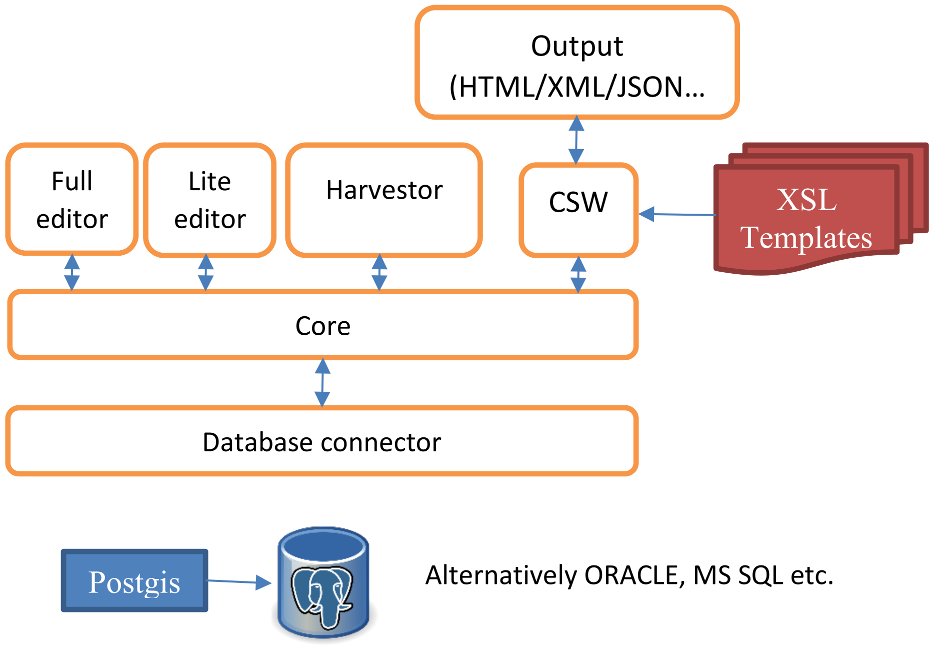


Figure 35: EGDI metadata catalogue (MIcKA) internal schema

4.5.2.2 Software components

MIcKA is a web application for management and cataloguing of metadata for structured data. It is based on the following technologies: PHP Nette framework, XLST, PostgreSQL fulltext search, JQuery, Bootstrap and OpenLayers. The data is stored in a dedicated PostgreSQL database that is backed up regularly once a week. The latest version of the backup is kept for a month.

Component	Type	Version
	Database backend	PostgreSQL 9.2.24 + PostGIS 2.0
	Http Server	Apache/2.4.23 (Linux/SUSE)
	Programming language	PHP 7.0.7
	Development framework	Nette Framework 2.4
	Frontend components	JQuery 1.11.1, Bootstrap 3.3.7, Select2 4.0.13, OpenLayers 5.0
	Operating System	Suse Linux 12 SP5

4.5.3 Metrics

4.5.3.1 Number of users

There are two basic categories of users – metadata editors and metadata browsers. The number of GeoERA metadata editors is expected to remain approximately the same and no more than 60. The number of unique visitors browsing the EGDI Metadata Catalogue was between 20 and 40 per month. The ratio between the new visitors and returning visitors was 27%:73%.

4.5.3.2 Number of requests

Users visited the catalogue with 50-90 visits per month, which represents an average of 300 pages per month. This number may vary over time and may increase slightly as GIP-P provides new data from other GeoERA projects. On the other hand, the search engine will help users to easily find their data of interest, which can reduce the number of browsing users.

4.5.4 User review

The development version of the MIcKA 6.0.305 application was published in June 2019. Then, testing was performed by comparing the results with previous SW versions. The test results verified that the requirements for INSPIRE availability and performance were met. Query tests were also performed using the INSPIRE API. Tests on the JRC validator for profile compatibility with the INSPIRE profile were problematic, as the JRC validator did not have some items updated according to INSPIRE 2.x at the time of testing.

4.5.2.3 1st round of functionality testing:

In June 2019, a request for functionality testing was sent to the GIP-P consortium. GEUS, IGME and CGS performed a test and submitted comments.

During July, August and September 2019, all comments were categorized and resolved. In September 2019, EGDI MIcKA version 6.1.404 was launched and filled with all metadata from EGDI MIcKA v. 5 (including the regular harvesting mechanism). Deliverable 7.1 Working version Metadatabase presents the results of the job.

4.5.2.4 2nd round of functionality testing:

In March 2020, the new version of EGDI MIcKA version 2020.010 (https://egdi.geology.cz) was debugged and extended with a new editing form EGDI-Lite. The latest version of the GeoERA keyword thesaurus located at the final address data.geoscience.earth was linked to the EGDI metadata catalogue. The Project Vocabulary was linked with a temporary address due to ongoing development.

The updated technical documentation was published and is available at https://czechgeologicalsurvey.github.io/MICKA-Docs/index.html:

1. Detailed technical documentation of the EGDI metadata profile, including a description of 3D models
2. Cookbook for creating metadata records using the EGDI metadata catalogue

The new version of the EGDI Metadata Catalogue defined the minimum required items for metadata. The user is not allowed to save a metadata record without filling in at least these 5 mandatory items (see table below). The Cookbook provides more detailed instructions for filling in the elements of the EGDI profile in the EGDI-Lite editing form for the spatial dataset.

Minimum mandatory items for metadata:

1	Resource title
2	Resource abstract
3	Resource type
19	Responsible party
28.1.	Metadata point of contact

The EGDI-Lite editing form is brand new and was created for GeoERA project partners to help them fill in metadata in a familiar way. This form is simpler and more userfriendly than the Full-editing form, which was so far the only one used for EGDI editors. Testing of the EGDI-Lite form was a major part of this testing phase during April.

In April 2020, 36 user accounts were created for the GeoERA projects metadata editors and a request for testing was sent to all together with the cookbook and EGDI metadata profile documentation.

Editors were asked to:

1. **test harvesting** from national or project metadata catalogues to EGDI, if they exist (to provide the address of the CSW service)
2. **insert a metadata record** directly to the EGDI Metadata Catalogue at least for the testing purposes
3. **send additional codelists**, if required

The results of testing the EGDI-Lite form, which was performed in April and early May, were processed. As of the date of this report, 38 GeoERA users have received their login details to the EGDI Metadata Catalogue. Some of them are going to join the testing later. A total of 11 editors from 10 projects responded and collaborated on testing. Unfortunately, we have not received any reaction from 5 projects.
Overall, users were satisfied with the new EGDI-Lite editing environment. Problems only occurred at the beginning of testing, when cloning and saving records. It was necessary to better warn the users that the required minimum of items must be filled. These errors were corrected immediately.
All other comments from the catalogue testing were reviewed, processed and divided into bugs, tasks and enhancements. Bugs were corrected in April/May 2020, the most important amendments and changes are being processed in June. Additional comments defined as enhancements may be included in the next version of the EGDI metadata catalogue.
The EGDI-Lite editing form, the Cookbook and the documentation will be modified accordingly. The Cookbook will be extended with a new chapter with detailed instructions for filling the EGDI profile elements in the EGDI-Lite editing form for services.
More detailed status and testing results are given in the following table (Y=yes, N=no):

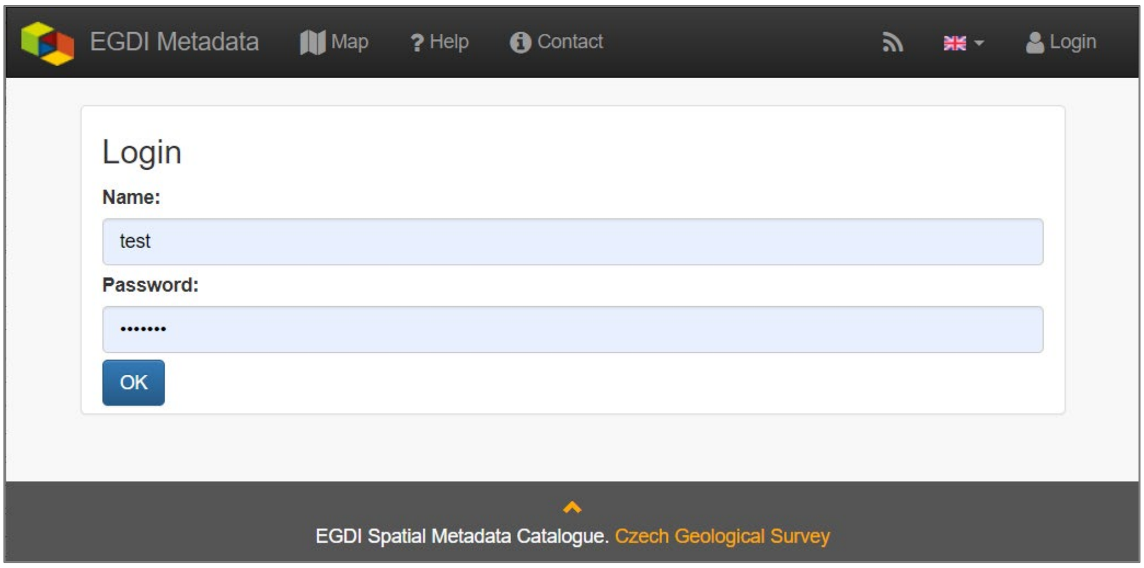
Projects	**responded (logged)**	**did test**	**created record**	**plan to harvest**	**harvesting status**	**sent comments**	**settlemen status of comment**
3DGEO-EU	Y	Y	Y			Y	done
HIKE	Y	Y	Y			N	
HotLime	Y	Y	Y			Y	in progress
MUSE	Y	Y	Y	Y	waiting for URL	Y	in progress
GeoConnect³d							

GARAH	Y	Y	Y			Y	done
HOVER	Y	Y	Y	Y	already set up	Y	done
TACTIC	Y (2 eds.)	Y (3 eds.)	Y (3 eds.)	Y	waiting for URL	Y (2 eds.)	done
RESOURCE							
VOGERA							
FRAME	Y	N, later					
MINDeSEA	Y	Y	Y			Y	in progress
EuroLithos							
Mintell4EU							
GIP	Y	Y	Y	Y	already set up	Y	done
Total	**10 Y**	**9 Y +**	**9 Y**	**4 Y**	**2 Y +**	**8 Y**	**5 Y +**

4.5.5 Envisioned scenario for performance testing

4.5.2.5 Stage 1

Type the following web address https://egdi.geology.cz/sign/in. Use a password and a username obtained on email request from administrator **egdi.metadata@geology.cz**.



Login

Name:

test

Password:

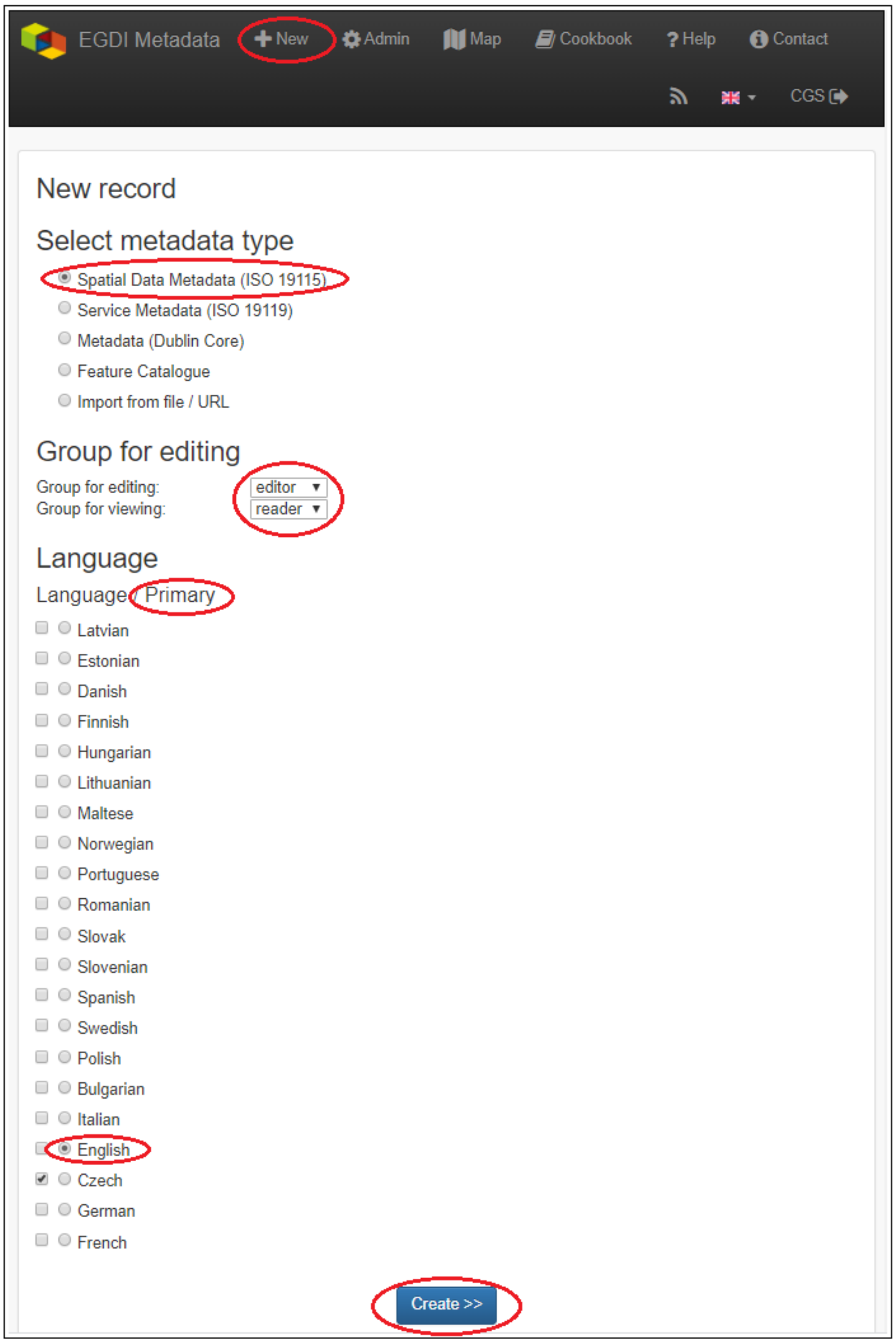
•••••••

EGDI Spatial Metadata Catalogue. Czech Geological Survey

Figure 36: Login page

4.5.2.6 Stage 2

Try to create new metadata record manually directly in the EGDI Metadata catalogue https://egdi.geology.cz/record/new. If in doubt, the cookbook will help you https://egdi.geology.cz/catalog/micka/cookbook.



EGDI Metadata · New · Admin · Map · Cookbook · Help · Contact · CGS

New record

Select metadata type

[✓ Spatial Data Metadata (ISO 19115)]
Service Metadata (ISO 19119)
Metadata (Dublin Core)
Feature Catalogue
Import from file / URL

Group for editing

Group for editing: editor
Group for viewing: reader

Language

Language / Primary

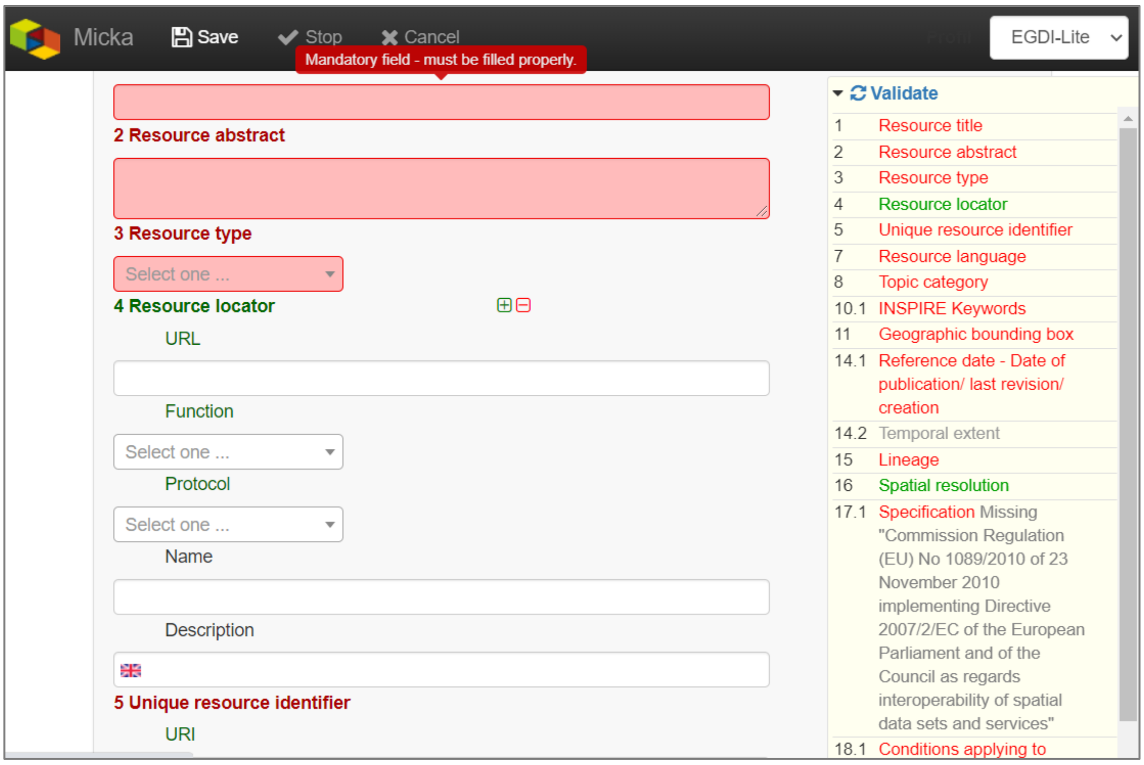
Latvian
Estonian
Danish
Finnish
Hungarian
Lithuanian
Maltese
Norwegian
Portuguese
Romanian
Slovak
Slovenian
Spanish
Swedish
Polish
Bulgarian
Italian
[✓ English]
✓ Czech
German
French

Create >>

Figure 37

4.5.2.7 Stage 3

Fill in all mandatory fields, which are coloured red to allow you to save the record.



Mandatory field - must be filled properly.

2 Resource abstract

3 Resource type

Select one ...

4 Resource locator

URL

Function

Select one ...

Protocol

Select one ...

Name

Description

5 Unique resource identifier

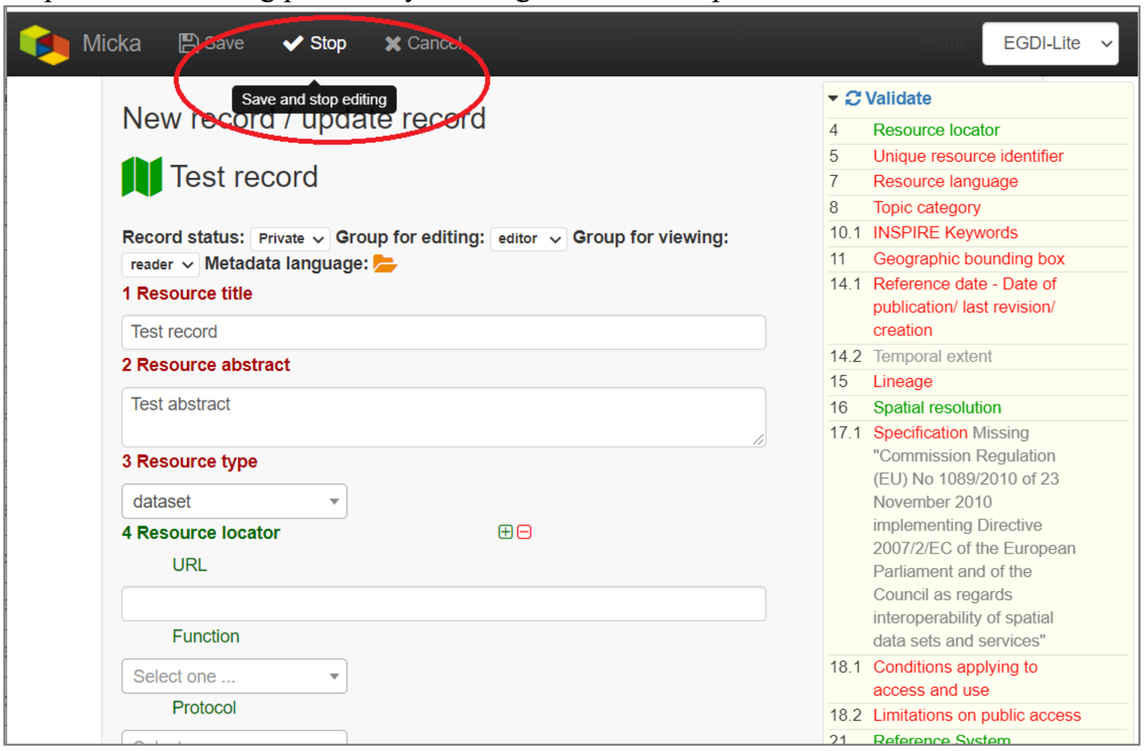
URI

1	Resource title
2	Resource abstract
3	Resource type
4	Resource locator
5	Unique resource identifier
7	Resource language
8	Topic category
10.1	INSPIRE Keywords
11	Geographic bounding box
14.1	Reference date - Date of publication/ last revision/ creation
14.2	Temporal extent
15	Lineage
16	Spatial resolution
17.1	Specification Missing "Commission Regulation (EU) No 1089/2010 of 23 November 2010 implementing Directive 2007/2/EC of the European Parliament and of the Council as regards interoperability of spatial data sets and services"
18.1	Conditions applying to

Figure 38: Mandatory fields must be filled

4.5.2.8 Stage 4

Stop and save editing process by clicking on button Stop.



New record / update record

Test record

Record status: Private Group for editing: editor Group for viewing: reader Metadata language:

1 Resource title

Test record

2 Resource abstract

Test abstract

3 Resource type

dataset

4 Resource locator

URL

Function

Select one ...

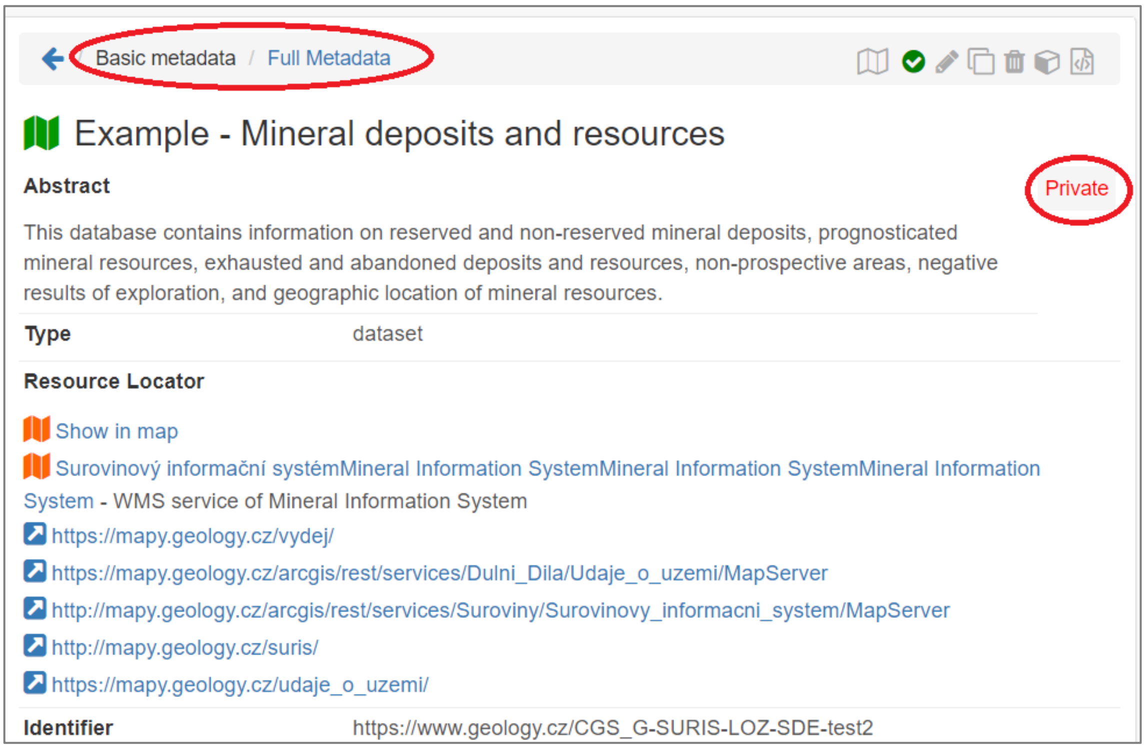
Protocol

4	Resource locator
5	Unique resource identifier
7	Resource language
8	Topic category
10.1	INSPIRE Keywords
11	Geographic bounding box
14.1	Reference date - Date of publication/ last revision/ creation
14.2	Temporal extent
15	Lineage
16	Spatial resolution
17.1	Specification Missing "Commission Regulation (EU) No 1089/2010 of 23 November 2010 implementing Directive 2007/2/EC of the European Parliament and of the Council as regards interoperability of spatial data sets and services"
18.1	Conditions applying to access and use
18.2	Limitations on public access
21	Reference System

Figure 39

4.5.2.9 Stage 5

The metadata record was created and it can be displayed as a basic or full metadata view. If you are satisfied with the metadata, but your record is still private, it is not visible to everyone.


Basic metadata / Full Metadata

Example - Mineral deposits and resources

Private

Abstract

This database contains information on reserved and non-reserved mineral deposits, prognosticated mineral resources, exhausted and abandoned deposits and resources, non-prospective areas, negative results of exploration, and geographic location of mineral resources.

Type	dataset

Resource Locator

Show in map

Surovinový informační systémMineral Information SystemMineral Information SystemMineral Information System - WMS service of Mineral Information System

https://mapy.geology.cz/vydej/

https://mapy.geology.cz/arcgis/rest/services/Dulni_Dila/Udaje_o_uzemi/MapServer

http://mapy.geology.cz/arcgis/rest/services/Suroviny/Surovinovy_informacni_system/MapServer

http://mapy.geology.cz/suris/

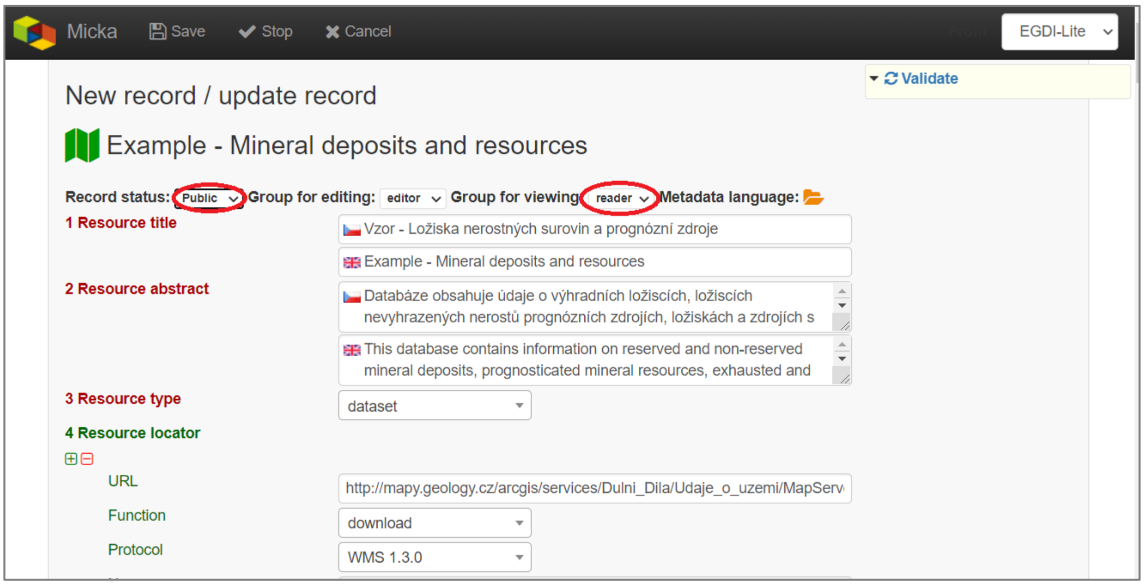
https://mapy.geology.cz/udaje_o_uzemi/

Identifier	https://www.geology.cz/CGS_G-SURIS-LOZ-SDE-test2

Figure 40

4.5.2.10 Stage 6

Make your record “public” and set Group for viewing to value “reader”, so that all users can search and view the record.



New record / update record

Example - Mineral deposits and resources

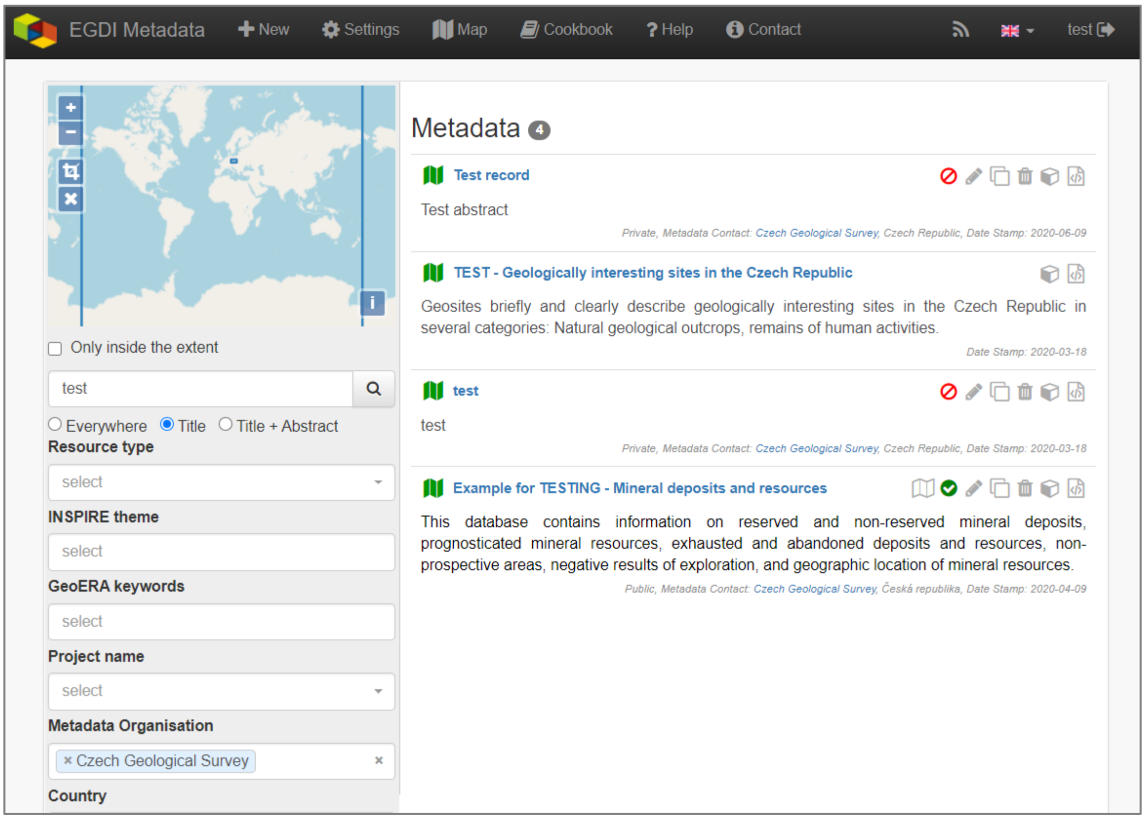
Record status: Public Group for editing: editor Group for viewing: reader Metadata language:

1 Resource title	Vzor - Ložiska nerostných surovin a prognózní zdroje
	Example - Mineral deposits and resources
2 Resource abstract	Databáze obsahuje údaje o výhradních ložiscích, ložiscích nevyhrazených nerostů prognózních zdrojích, ložiskách a zdrojích s
	This database contains information on reserved and non-reserved mineral deposits, prognosticated mineral resources, exhausted and
3 Resource type	dataset
4 Resource locator	
URL	http://mapy.geology.cz/arcgis/services/Dulni_Dila/Udaje_o_uzemi/MapServ
Function	download
Protocol	WMS 1.3.0

Figure 41

4.5.2.11 *Stage 7*

Go to the homepage https://egdi.geology.cz/ and search any metadata you need.



Metadata 4

Test record

Test abstract

Private, Metadata Contact: Czech Geological Survey, Czech Republic, Date Stamp: 2020-06-09

TEST - Geologically interesting sites in the Czech Republic

Geosites briefly and clearly describe geologically interesting sites in the Czech Republic in several categories: Natural geological outcrops, remains of human activities.

Date Stamp: 2020-03-18

test

test

Private, Metadata Contact: Czech Geological Survey, Czech Republic, Date Stamp: 2020-03-18

Example for TESTING - Mineral deposits and resources

This database contains information on reserved and non-reserved mineral deposits, prognosticated mineral resources, exhausted and abandoned deposits and resources, non-prospective areas, negative results of exploration, and geographic location of mineral resources.

Public, Metadata Contact: Czech Geological Survey, Česká republika, Date Stamp: 2020-04-09

Only inside the extent

test

Everywhere [✓ Title] Title + Abstract

Resource type: select

INSPIRE theme: select

GeoERA keywords: select

Project name: select

Metadata Organisation: Czech Geological Survey

Country

Figure 42

4.5.6 Scalability issues

The EGDI metadata catalogue uses the MIcKA system for management and publication of metadata on structured data. MIcKA 6 technology enables entry, editing, harvesting, discovery, and view of metadata on geological data across Europe. It provides tools for compilation and export of the metadata in a standardized format.

4.5.7 Performance testing

No performance test has been performed since the new software version was implemented. Testing will be performed later when the system is fully loaded. The servers are designed for increased load.

4.5.8 Planned improvements

During testing, several comments were collected on the EGDI-Full editing form used by experienced users of the EGDI Metadata Catalogue. This editing form is necessary to fill in some additional information, such as special types of keywords required by 3D data providers. Therefore, for the next period of testing and development, we will focus primarily on debugging this editing profile and creating its detailed documentation. The sustainable development of the catalogue is inevitable.

4.6 Extensions to the harvesting system

4.6.1 Introduction

This document describes the Minerals data harvesting (M4EU harvesting) system provided by GeoZS and the whole harvesting process which includes data providers. The purpose is to harvest the mineral data from multiple providers into one central database.

Usually each country has its own specific way to describe geospatial environmental data. So, you have to involve the Infrastructure for Spatial information in Europe (INSPIRE) EU initiative to solve this problem. INSPIRE defines common standards to describe and share the spatial data across borders, requires that data is compatible across borders and that web services are used for data distribution. Web Feature Service (WFS) provides standard interface, allowing requests for geographical features across the web by using platform-independent calls.

The Geological Survey of Slovenia (GeoZS) implemented a harvesting system to collect and validate INSPIRE compliant spatial European geological data distributed through WFS 2.0 standard services.

Each data provider has to establish his own database and services on it. The Geological Survey of Slovenia (GeoZS) runs harvesting on provider request or periodically to collect data from all data providers in the central European geological database.

4.6.2 Architecture

The whole harvesting process is divided into provider (country) sites and the European level.

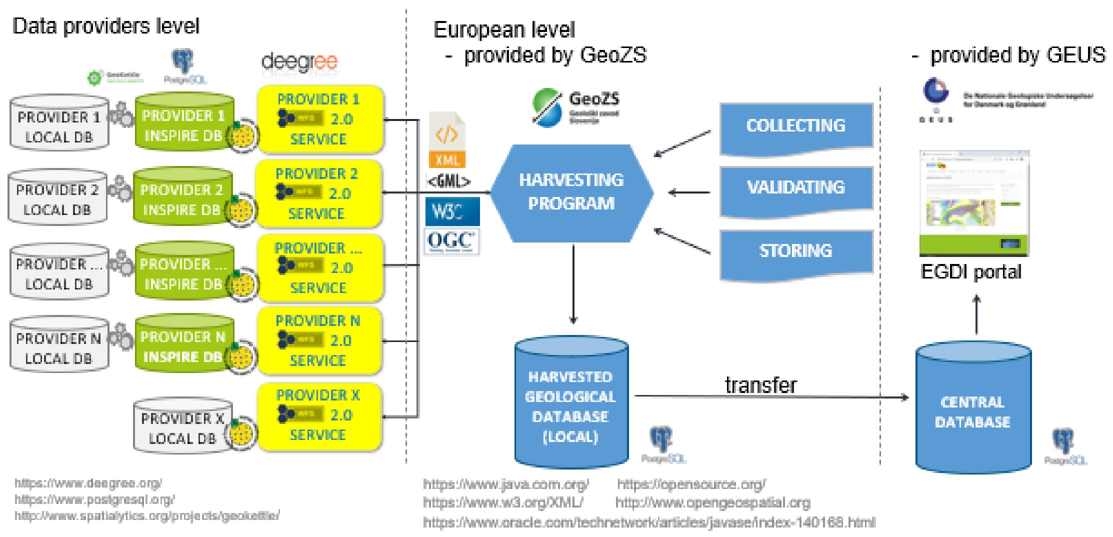


Figure 43: Harvesting architecture

4.6.2.1 Software modules and version

Module	Side	Type	Version
Java	EU/GeoZS	*programming language*	OpenJDK Runtime Environment 1.8.0
	EU/GeoZS	API library	JAXB API 2.3.x
Apache NetBeans	EU/GeoZS	Developmente nvironment	Apache NetBeans IDE 11.0
PostgreSQL	EU/GeoZS & Provider	Database Manager	PostgreSQL 10.10 + PostGIS 2.4.4
Apache Tomcat	Provider	Aplication Server	Apache Tomcat 8.x

deegree	Provider	Geospatial data management	Deegree 3.x
GeoKettle Open Source Spatial ETL	Provider	Extract, Transform and Load) tool	
or Windows Server	EU/GeoZS & Provider	Operating System	Independent

4.6.3 Metrics

4.6.3.1 Number of expected users

Currently we collect data from 29 providers. In the near future we expect two more. The harvesting system is not intended for public use. Therefore, the person responsible for performing harvesting runs it when he or she is notified by providers that there are new data available (or the existing data have changed) or periodically (usually on a monthly basis).

4.6.3.2 Number of requests

It mostly depends on the number of data providers (29 at the moment) and the number of data in their databases. The requests are made to different WFS services. We currently estimate approximately 1 million requests per harvesting.

4.6.4 Envisioned scenario for performance testing

4.6.4.1 On the data provider side

Each data provider has its own database with data which is compliant with the INSPIRE directive. The database also contains tables/views on which Deegree WFS 2.0 services are based. The data provider is responsible for maintaining his data in the database and that his services are available.

4.6.4.2 On European level – provided by GeoZS

At GeoZS we run the harvesting process periodically or on provider’s request. The harvesting program calls services on provider side using a GET request. The response is in XML format which is then parsed using the Java programming language and the JAXB library. The parsed data gets collected, validated and stored into an appropriate table in a central database. The system is implemented in Java with the use of parallel computing features and the whole solution is supported by open source software. After the harvesting process is finished for all countries, the records are checked in a way to prevent data gaps regarding missing data from some provider(s). If data gaps are detected, then we communicate with the provider to solve the problem. Otherwise we use the last successful harvesting data for that provider (provider referential database). Finally, a copy of the database is sent via ftp to the EGDI central database at GEUS from where data can be shown on the EGDI web GIS.

4.6.5 Scalability issues

The harvesting process is scalable because it harvests data in parallel for every provider. New providers can be added easily.

4.6.6 User review

The test harvesting is performed when the data provider does some environment changes (like migrating their software on different hardware, software upgrades, ...), bigger changes in data entries, when the programming code is improved or when errors are detected.
Data providers will compare data in harvested database tables with data in their original database tables and report all discovered differences.

4.6.7 Performance testing

So far there have been not outage of the harvesting system either on the data provider side or the harvesting side. The metrics so far do not reflect any needs for a higher usage. Therefore performance testing has not been judged useful at that time.

4.6.8 Planned improvements

For harvesting M4EU v 2.04 database the following improvements are planned to be implemented:

- **To implement a solution to removing a data gap in the harvesting process**: the harvested database must have no data gaps. Currently, when the harvesting process is run, it is not rare that harvesting of some provider's data fails. These failures happen, for example, when the provider's servers are not online, when the server is too busy and not responding correctly, due to errors because of the incorrect data entry by the provider or owing to errors because the provider modifies its data by following a ETL process at the harvesting time. In the occurrence of any failures listed in this paragraph, the system will keep the last successful harvesting data for those providers.
- **To implement a report for a quick checking of the harvesting results**: after harvesting is finished, currently, only the database dump is provided. This is insufficient; it should provide a report with an overview of the records harvested by the providers. In that report, the dates for provider harvesting should be shown.
- **To implement a solution for geometry checking**: if the provider enters the geometry data incorrectly, the data cannot be shown on the map. We need a tool for geometry checking and reporting irregularities.

- **To provide a support for more than one data provider per-country**:
  because one country could have more than one provider, we need to modify m4eu database to add a country code and/or data provider acronym name to each table. In that way, we can assure the correctness of the data source
- **To provide a solution for minerals service monitoring**:
  currently, providers do not know if some of the required services are up or down. The harvesting process fails if the services are not up. By providing a service monitoring, the provider will immediately be aware of the problem and will be able to fix it. With the help from the monitoring services, the harvesting process will not start if services are not up and running
- **To provide a better logging**:
  currently, the harvesting process has one big log file, with a non-transparent content. With the current logging, it is impossible to see the history of logging for a provider. A better and more transparent logging is desirable for each data provider
- **The improvement of the current harvesting code**:
  the harvesting results are sometimes strange or some data appear not to be harvested. In order to use the data, EGDI needs complete and correctly harvested datasets. Therefore, an extensive testing, discovering bugs, a code debugging and fixing is required for the harvesting process. Improvements in the harvesting code should also include all requested improvements in the database, code lists, views, services, etc.
- **To provide tests with providers**:
  providers must help to improve the harvesting process by comparing their source database with the harvesting results and report all identified differences to harvesting@geo-zs.si. They must also send information on data change in their databases. Any missing tables, differences in records count and different field values must be discovered and corrected.
- **To check for stable INSPIRE ID fields**:
  an identifier should correspond always to the same deposit and should not be changed by the provider. Providers should normally change just the version ID and the begin/end life span. Providers usually change the data by ETL process, which first deletes all their data in their database and create a new set. In that process, providers generally do not take into consideration stable INSPIRE id fields. We cannot just trust that stable INSPIRE ids where provided. We need an automatic mechanism to check and report deviations, so checking should be added to the harvesting process in some way.
- **To provide a history information for number of data records for provider**:
  To track a progression in amount of data provided by a provider, we need a solution for showing the history of the number of harvested records from the m4eu database tables. This could be solved by creating a special table or database with the harvesting history data (history of count reports), which could provide us a progression view for a certain provider (history of number of data entries in each table).
- **Option for recreating a provider local database**:
  That is, to provide a solution to data providers, so they can recreate their

harvested database on their side. This solution will provide an additional option to providers to check if their database was harvested correctly.

- **Add m4eu version log table with service**:
  Information about the m4eu database version, code list version and views version. Maybe, we could also have a record where users can put their last database modification date. All that information must be provided by services. That information can be checked by the harvesting process.
- **To provide an information for the latest database modification by the provider**:
  It is not important only when the last harvesting was provided by the country, but also when a provider modifies the database. Providers will be required to manually change some fields in their database after each data update.
- **The integration of the next version of M4EU database**:
  The ORAMA project gave some recommendations to improve the M4EU database, the software versions of Java Development Kit (open source), Apache Tomcat, Deegree3, and PostgreSQL and the mapping of the WFS-services. Many of the ORAMA recommendations have now been tested, but some still need to be tested before they are implemented in the next version.
- **Differences between current M4EU DB v1.1.2 and next version v.2.x**:
  - 2 new DB tables 'commoditygrouptype' and 'totalproduction'
  - Updates of more codelists tables especially tables 'commoditytype', 'UNFCategoryType' and 'WasteTypeType'
  - Adding myb schema with tables, views and codelists for Mineral Year Book data
  - (nice to have but currently not planed: Updates of INSPIRE schemas)

4.7 Administration module

4.7.1 Introduction

The EGDI Administration module is a *web site* allowing users (GIP-users and selected project-members) to upload, edit and delete content which be available in the "Web GIS" and in "The EGDI document repository search thematic application".

From a user-perspective, the Administration module comprises of the following key functionalities:

- Create *maps* to be shown in Web GIS (showing data from created *data sets*)
- Create *data sets* by uploading spatial data files:
  - Geopackage files
  - ESRI Shape files
- Upload files and associate common metadata
  - Upload jpeg files
  - Upload pdf files
  - Upload csv files (not implemented yet)
  - Upload doi links
- Edit uploaded files and their metadata

- Login functionality (so users can only edit/upload files or datasets within their own project)

The system is a full-stack system using client-side logic (browser), server-side logic, and data storage.

Client-side logic	Interactive web page (JSP and JavaScript) with embedded logic using common browser modules like jQuery, Bootstrap. Servered by Tomee
Server-side logic	A Java (JSP) web application running in Tomee. Uses various jar libraroes to read metadata from jpg, pdf files. Uses mapserver/gdal to import shapefiles and GeoPackage files into the database (as data sets).
Storage (database & file)	The application uses a PostgreSQL database to store and retrieve maps, data sets and metadata. The database also holds user credentials. The uploaded files are stored in a file share on the server.

4.7.2 Architecture

4.7.2.1 Production environment

The production version of the Administration module is currently installed on the same server as the Web GIS module. The server name is geusegdi01 (see section 1.8.2 for details). The url for the modules web interface will be located at https://admin.europe-geology.eu.

The most current installation instructions can be found in GitLab at: https://geusgitlab.geus.dk/egdi/egdiadmin/-/raw/master/docs/install_egdiadmin_on_linux.txt

4.7.2.2 Test environment

The systemtest environment is used for

- Integration test between different modules (so we know that they work together)
- Testing of new features by the users (requirement testing and usability testing)
- Practicing use of the features (upload etc)

The test environment is used for feature testing and usability testing of the Administration module. The url for the web interface will be located at https://admin-

test.europe-geology.eu.
The test environment runs on the server called egditest01 and uses identical hardware as the production environment.

4.7.2.3 Architecture diagram

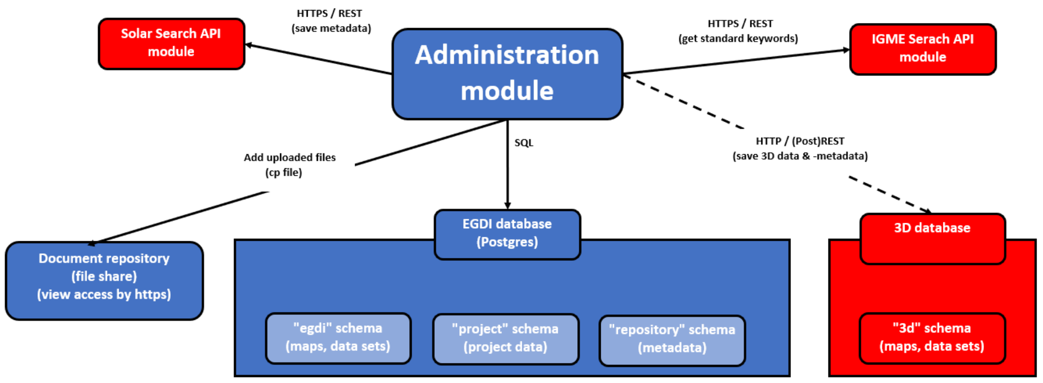


Figure 44: Administration module arhitecture

In system architecture diagram shown above you can see:

- The integrations of the different modules that the Administration module consist of (shown in blue)
- The communication of the Administration module with other modules (shown in red)

4.7.3 Metrics

4.7.3.1 Number of users

We expect a few simultaneous users 1-10 and only a few 1-3 doing heavy operations (upload of spatial data) at the same time.

4.7.3.2 Number of requests

A low number due to the few expected users. The system should be able to handle 5 GET page requests pr. second.

4.7.4 Envisioned scenario for performance testing

The Administration module has been tested by the “test scenarios”, “end to end flow test” and “manual test” method mentioned in section 3.1. No automatic unit tests or integration tests have been implemented.

Suggested end-to-end flow tests

a) Upload pdf file with no existing metadata

- Upload a pdf file (without any metadata in the file)
- Enter the metadata
- Verify that the file is uploaded correct
    - That the pdf url is working
    - That metadata looks correct in the administration module (in the "docs" page)
    - That metadata looks correct in solar
    - That metadata is written to the pdf file (download the file and view in acrobat)

b) Upload pdf file with existing metadata

- Upload a pdf file (with metadata already present in the file)
- Verify that you can see the (existing) metadata in the GUI
    - Overwrite some of the metadata
- Verify that the file is uploaded correct
    - That the pdf url is working
    - That metadata looks correct in the administration module (in the "docs" page)
    - That metadata looks correct in solar

c) Upload jpg file with no existing metadata

- Upload a jpg file (without any metadata in the file)
- Enter the metadata
- Verify that the file is uploaded correct
    - That the url is working
    - That metadata looks correct in the administration module (in the "docs" page)
    - That metadata looks correct in solar
    - That metadata is written to the jpg file (download the file and view in acrobat)

d) Upload jpg file with existing metadata (including gps position)

- Upload a jpg file (with metadata in the file – title, gps position)
- Verify that you can see the (existing) metadata in the GUI
    - Overwrite some of the metadata
- Verify that the file is uploaded correct
    - That the url is working
    - That metadata looks correct in the administration module (in the "docs" page)
    - That metadata looks correct in solar
    - That metadata is written to the jpg file (download the file and view in acrobat)

e) Upload geopackage file with multiple layers

- Upload a geopackage file (with at least two layers)
- Verify that all layers are upload in the GUI
  - Verify that you can see the preview of all the layers

f) Upload a shapefile

- Upload shapefiles (with at least two layers)
- Verify that all layers are upload in the GUI
  - Verify that you can see the preview of each of the layers

<u>Suggested scenario tests</u>

a) Login to your project (user with a single project)

- Try to login to the administration module with a user having access to one project (etc HIKE)
- Enter username
- Enter password
- Verify that you can login (the project will be shown at the right top of the screen)

b) Login to your project (user with multiple projects)

- Try to login to the administration module with a user having access to multiple projects (etc HIKE and TACTIC)
- Enter username
- Enter password
- Verify that you can login (if you select a project (etc. HIKE))

c) Reject login if wrong project (user with no access to project)

- Login to the administration module with a user with no access to a project (etc HIKE)
- Enter username
- Enter password
- Select project (etc. HIKE)
- Verify that you cannot login (get error message)

4.7.5 Scalability issues

Some of the features in the EGDI Administration module relies on heavyweight system operations such as importing spatial data files (etc. GeoPackage files) through mapserver / gdal and processing metadata in the uploaded files (etc pdfs).

The module is therefore not expected to scale to a lot of simultaneous users - if they are using these heavyweight operations at the same time. This is ok given that the module is intended for a few expert users who occasionally needs to upload or edit content. It is intended to test how many users can do these heavy weight operations at the same time.

If a higher number of simultaneous users is desired the system can be scaled either horizontally (by increasing RAM and disk IO) or verticallyby running serval instances of the Administration module (etc. using docker images) with a load balancer in front(see section 4.1.5.1 regarding load balancer for the WebGIS) .

4.7.6 User review

After the Administration module was released to systemtest the module has been tested by selected GSP users from the HIKE, HOVER and TACTIC projects.

The users have been given an account on the test system and are asked to upload documents which would be used in their projects. Feedback was received by mail.

List of performed GSP user tests:

Date	Project	Tested	Remarks
02/04-2020	HIKE	Login Upload PDF	Secure PDFs must also be supported in upload (suggested some UI changes)
16/04-2020	HIKE	Login Upload secure PDFs UI changes	Tested ok
20/05-2020	HOVER	Login Upload Pdf	PDF upload is easy to use (requested for case insensitive login (already implemented))
11/06-2020	TACTIC	Login PDF upload JPG upload	Tested ok

4.7.7 Performance testing

All the tests described in “end-to-end-flow-tests” and “scenario tests” have already been performed manually by the development team (GEUS). Also a lot of manual tests have been performed.
All of the described “end-to-end-flow-tests” have also been carried out by selected end users of the “TACTIC”, “HIKE” and “HOVER” projects. Their suggestions and found issues have already been implemented (case incentive login of username, upload of protected pdfs).

Because features are being added to “the administration module” continuously (like DOI uploads). The list of test scenarios and end-to-end-tests is expected to expand in the future. Also, it is recommended to retest all of the flows as new features might affect already implemented features.

4.7.8 Planned improvements

In the future it is planned that the other GSP also test these feature – so we know the implemented features for upload of documents are adequate for all the projects. Also the upload of the following features should be user tested / reviewed:

- DOI upload (upload a document by doi reference)
- CSV upload (upload csv data files)
- Upload of geo package data
- Upload of shape files

4.8 Vocabulary tool

4.8.1 Introduction

Project Vocabularies are collections of controlled dictionaries containing essential information about scientific concepts relevant for a project. The primary goal is to support projects and datasets with linguistically labelled terms. Project Vocabularies provide stable and reusable links to concepts (units of thoughts) that can be referenced whenever unambiguity is important. Behind such links alternative names, translations, definitions synonyms and additional information about other related concepts are made available. In any situation when something must be unambiguously named, a concept from a Project Vocabulary can be used. A Project Vocabulary can facilitate search and information access in a linked data environment. The GIP-P thus encourages all projects to create projects vocabularies.
After comparison and thorough test the EGDI platform has selected the UKGovLDregistry[13]. It has been installed and is accessible at http://data.geoscience.earth/ncl/.

[13] https://github.com/UKGovLD/registry-core

4.8.2 Architecture

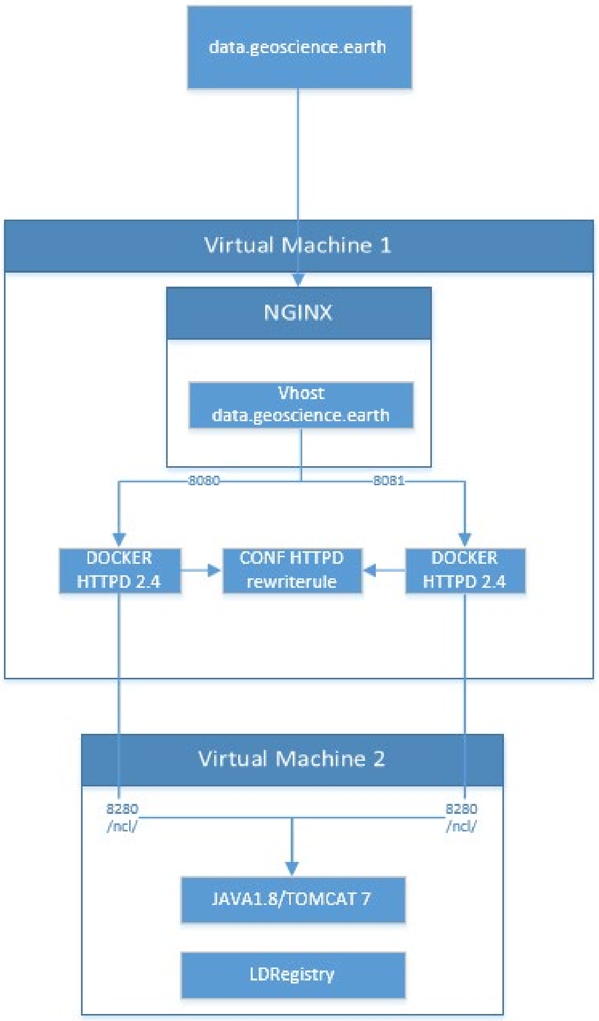


Figure 45 : Vocabulary tool architecture

4.8.2.1 Hardware

Module	CPU	RAM
VM 1	2	4Go
VM 2	1	4Go

4.8.2.2 Sofware

Module	Type	Version
VM 1	Operating System	Linux – Centos 7
VM 2	Operating System	Linux – Centos 7
UkGovLDregistry	Vocabulary tool	

4.8.3 Metrics

Metrics are captured by using BRGM Motomo deployment at https://wwwstats.brgm.fr.

4.8.3.1 Number of users

There are about 15 to 25 users per day.

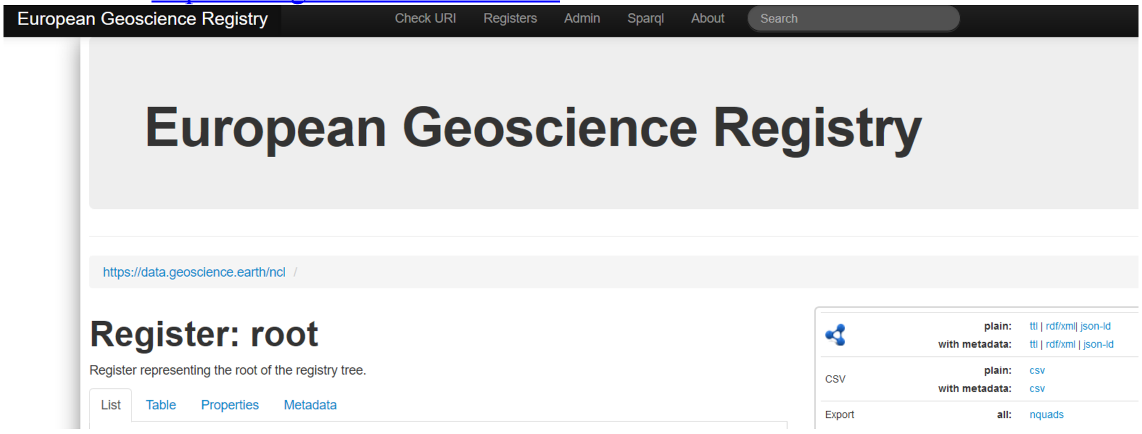
4.8.3.2 Number of requests

There are about 280 to 400 requests per day.

4.8.4 Envisioned scenario for performance testing

4.8.4.1 Stage 1

Empty your browser cache
Connect to http://data/geoscience.earth/ncl/



Register: root

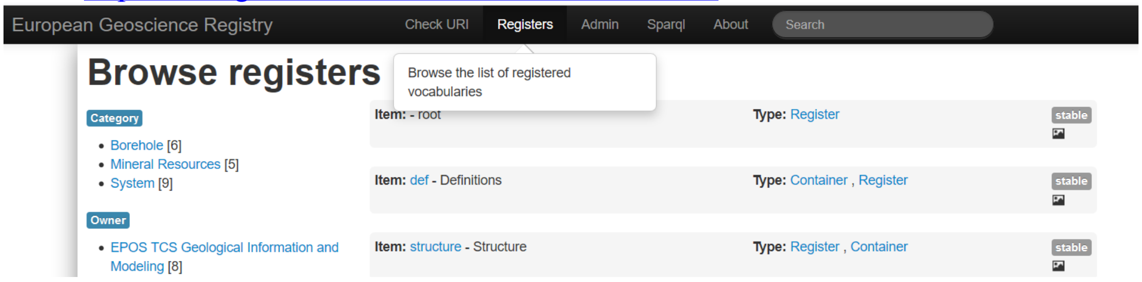
Register representing the root of the registry tree.

	plain:	ttl | rdf/xml| json-ld
	with metadata:	ttl | rdf/xml | json-ld
CSV	plain:	csv
	with metadata:	csv
Export	all:	nquads

Figure 46

4.8.4.2 Stage 2

Go to https://data.geoscience.earth/ncl/ui/dataset-search


European Geoscience Registry
Check URI
Registers
Admin
Sparql
About
Search
Browse registers
Browse the list of registered vocabularies
Category
Owner

- Borehole [6]
- Mineral Resources [5]
- System [9]
- EPOS TCS Geological Information and Modeling [8]

Item: - root	**Type:** Register	stable
Item: def - Definitions	**Type:** Container , Register	stable
Item: structure - Structure	**Type:** Register , Container	stable

Figure 47

4.8.4.3 Stage 3

Go to https://data.geoscience.earth/ncl/ui/sparql-form

4.8.4.4 Stage 4

Run the following SPARQL query which is used for harvesting all the basic terms without hierarchy.

```
prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
prefix owl: <http://www.w3.org/2002/07/owl#>
prefix xsd: <http://www.w3.org/2001/XMLSchema#>
prefix dct: <http://purl.org/dc/terms/>
prefix foaf: <http://xmlns.com/foaf/0.1/>
prefix skos: <http://www.w3.org/2004/02/skos/core#>
prefix version: <http://purl.org/linked-data/version#>

prefix reg: <http://purl.org/linked-data/registry#>


SELECT ?entity ?label
where {
values ?reg {<https://data.geoscience.earth/ncl/geoera/keyword>}
?item reg:register ?reg ; version:currentVersion ?itemVer.
?itemVer reg:definition ?x.
?x reg:entity ?entity.
?entity skos:prefLabel ?label.}
```

4.8.5 Scalability issues

The actual architecture of https://data.geoscience.earth is the result of an identified scalability issue. There was a bottleneck in the URI resolver. That has been solved by having multiple Apache Httpd server instance.
We have leveraged Docker containers, and because we foresee a more important usage of the registry in the future we envision the possibility to migrate to container orchestration with Kubernetes.

4.8.6 User review

The tool has not been developed in the project. It has been chosen based on requirement made by ontologists at BRGM. They thought that rather than reinventing the wheel they would better use an existing software and make requests for improvement.
As of now the ontologists are confident in using the tool and feel it fits its purpose.

4.8.7 Performance testing

The architecture so far is able to sustain the actual users of the system. We envision to test the performance of the vocabulary tool once we will have clearer use cases that could, for example, make a huge usage of linked data through scripting or AI.

4.8.8 Planned improvements

The planned improvements are done by the user community of the tools and are the following:

a) Notification mechanism for any action on a registry (Pub/Sub)
b) Support of the requested format for the registry federation INSPIRE
c) Internationalization of the interface (i18n)
d) Semantic consistency tests when adding entries to registers
e) Refinement of the life cycle management

4.9 Monitoring system

4.9.1 Introduction

The monitoring system lists and monitors the map layers that are registered on the EGDI portal.

The web application has the following functionality

- All layers/services in the EGDI database that are linked to maps on the EGDI portal have been registered in the Zabbix system running at GEUS. The registration is done via the Zabbix API (create_zabbix_from_egdi_layers.html). Zabbix will call getCapabilities for each registered service every hour and will log if the service returns an XML document and how fast the service answers.
- The main monitoring page (MonitorSmiley.html) presents the layers in a hierarchy table with two levels "Maps" and "Groups" corresponding to the layout on the EGDI portal. The two level can be collapsed in the visual layout by clicking on them with the mouse.
- For each layer it will be shown if there is registered metadata in the MICKA database. If there is metadata, then a green smiley is shown.
- For each layer an interpretation of the latest data from Zabbix will be shown. Red smiley if the service does not respond or does not respond with an XML document. Yellow smiley if the service takes a long time to respond. Green smiley if the service responds with an XML document within one second.
- One or more of the layers can be selected and then shown in the EGDI map viewer.
- When a specific layer is clicked then a new page (MonitorSmileyLayer.html) will open showing detailed information about the layer. For example a description, data from Zabbix, Links to the service and to the metadata.

4.9.2 Architecture

The monitoring system is running on this address: https://data.geus.dk/egdi_monitor_smiley/MonitorSmiley.html

The user interface consists of two Javascript web pages. The backend is a Zabbix installation and a Postgrest interface to the EGDI database. Both of these are called directly from the web pages.

4.9.3 Scalability issues

The monitoring systems is implemented with direct calls from the web page to the Zabbix server. Two calls are made for each layer that is monitored. The calls are made asynchronously for performance. The Zabbix status for 100 layers takes about one second to load. This delay when loading the page is considered reasonable.

4.9.4 User review

The design and functionality of the monitoring system have been shown to users at project meetings.

4.9.5 Planned improvements

The feature requests below have not been implemented yet

- A quality assessment of the metadata in MIcKA should be shown. Probably as a red, yellow or green smiley. How to do this can be discussed with MIcKA. A suggestion is to use INSPIRE compliance as the way to get a green smiley.

- It should be shown how well the service complies with the FAIR principles. What are the legal restrictions when using the service for example? This information is probably also something that should be extracted from the Metadata in MIcKA.
- More information about the services should be shown. Which coordinate systems are supported? Which output format are supported? Which OGC versions are supported? Do WFS services provide a rendering specification?
- It should be possible to download data files that have been delivered to EGDI (relevant for static data).
- Then Zabbix registration should call getMap instead of getCapabilities. This will answer more precisely if the service is running correctly.
- It should be possible to group the layers according to the project they are generated from in addition to the existing grouping by topic.
- The detail page should show the result of a getMap call (an image) to the service.
- When opening layers on the EGDI map the map should be zoomed in to the maximum extent of the layers.

5.SYNTHESIS

5.1 Global architecture

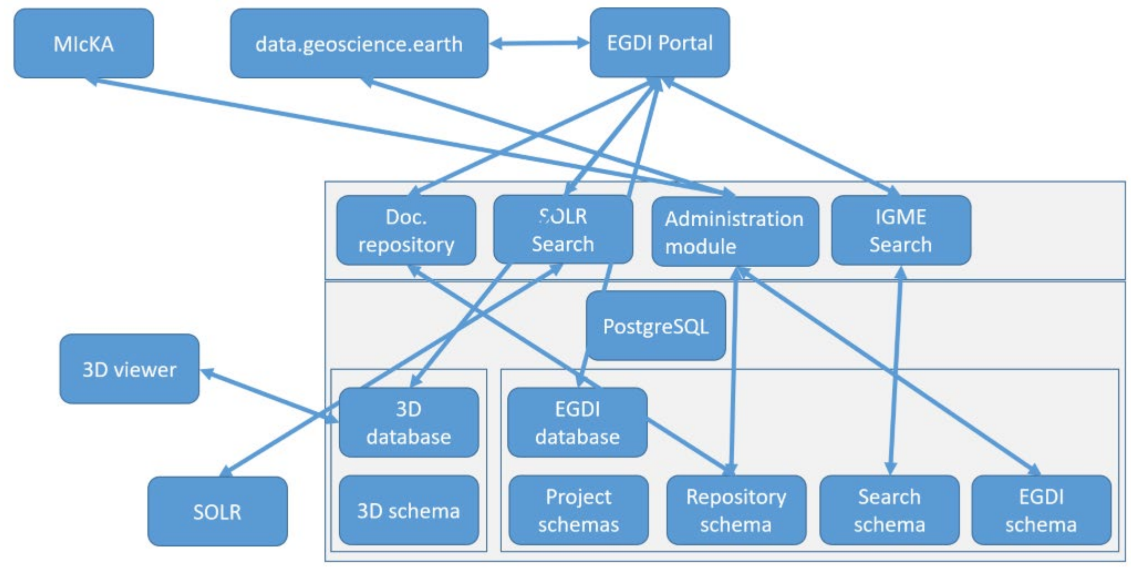


Figure 48 : EGDI Platform and modules global architecture

5.2 User review

All the module teams have identified the process for reaching the user and dedicated time to review their developed features.

5.3 Prototypes

Prototypes have been scheduled taking into account WP5. An architecture is being spawned on the development architecture. They will be evaluated later.

5.4 Static code analysis

In the GIP-P context, teams are not familiar with static code analysis. Hence, we were not able to assess the quality of the code. We plan to have dedicated small meetings, taking the shape of training session for spreading the use of automated static code analysis.
This will be performed using one of the most used open-source software: https://www.sonarlint.org which allows real-time analysis as the developer code.

5.5 Unit test policy proposal

Unit test has been started in the GIP project context. We propose the following approach.

5.5.1 Controllers test / WS

Unit tests should not only test the service called by a controller but also the controller itself. This makes it possible to check the correct retrieval of the WS parameters on a GET, POST or other and thus to check compliance with the WS interface contract.
The example of a @WebMvcTest annotation of the Spring framework allows to test, to instantiate a controller and to really test the call to the WS.
Example:

```
@RunWith(SpringRunner.class)@WebMvcTest(controllers = FicheController.class)public class
CreerFicheControllerTest { @Autowired private MockMvc mockMvc;  @Test public void
rechercherEntiteById() throws Exception {     this.mockMvc
.perform(post("/rest/private/rechercher/entite")   .header(HttpHeaders.AUTHORIZATION,
getAuthorizationHeader())   .param("gid", "7291")   .accept(MediaType.APPLICATION_JSON))
.andExpect(status().isOk())  .andExpect(jsonPath("$.gid").value(7291))
.andExpect(jsonPath("$.entite").value(ENTITE))
.andExpect(jsonPath("$.denomination").value(DENOMINATION_ENTITE))
.andExpect(jsonPath("$.inclue").value(ENTITE_PARENTE));}
```

5.5.2 Context for taking the tests

Each test must be repeatable indefinitely and always produce the same result. For this purpose, the environment required to run the test must be the same at the beginning of each run. A test context (test fixture[14]) must be set up.
The tests must be able to run on any environment (developer workstation, continuous integration platform, etc.) and must not depend on external resources to be installed beforehand (database, ldap, etc.).
Frameworks allow the creation of an autonomous and constant environment, similar to the target environment in order to be able to run unit tests.
- Database: The H2 database allows data to be stored in memory, does not require installation and is volatile. H2 allows the use of classical spatial functionalities.
- Object mock: Mockito allows to replace the input objects of the functionality to be tested by simulated objects whose behaviour and values are controlled.
- Web services mock: Wiremock allows to replace, in a transparent way for processing, the web services needed by the unit test by simulated web services whose behaviour and returned data are controlled.
Example:

```
@Rulepublic WireMockRule wireMockRule = new WireMockRule(8090); @Beforepublic void init() {
wireMockRule.stubFor(  any(urlMatching("/referentiels/v1/interlocuteur.xml\\?offset=[0-9]+&limit=[0-
9]+&derniereDateDeMAJ=2018-11-01"))  .willReturn(   aResponse()
.withBodyFile("ws_interlocuteur_par_type.xml")    .withHeader("Content-Type",
MediaType.APPLICATION_XML)    .withStatus(200)));      wireMockRule.stubFor(
any(urlMatching("/referentiels/v1/interlocuteur.xml\\?offset=[0-9]+&limit=[0-
9]+&derniereDateDeMAJ=2018-11-04"))  .willReturn(   aResponse().withStatus(204)));}
```

1.SMTP server: Green Mail is an SMTP server that does not require installation. It is instantiated in memory which makes it particularly suitable for testing the sending and receiving of mail in unit tests.
Example:

```
@Injectprivate MailConfiguration mailConfig; @Beforepublic void init() { smtpServer = new
GreenMail(new ServerSetup(25, null, "smtp")); smtpServer.start();} @Afterpublic void tearDown()
throws Exception { smtpServer.stop();}       @Testpublic void testBatchAbort() throws Exception {
String contenuMsg = null, mailSubject = null; Address[] mailFrom = null, destinataires = null;
...execution du traitement...  MimeMessage[] receivedMessages = smtpServer.getReceivedMessages(); if
(receivedMessages.length != 0) {  MimeMessage mail = receivedMessages[0];  contenuMsg =
GreenMailUtil.getBody(mail);  mailFrom = mail.getFrom();  mailSubject = mail.getSubject();
destinataires = mail.getAllRecipients(); }  assertThat(receivedMessages.length).isEqualTo(1);
assertThat(contenuMsg).isNotBlank(); assertThat(contenuMsg).contains("Une erreur s'est produite dans
le traitement"); assertThat(mailFrom).isNotEmpty();
```

[14] https://en.wikipedia.org/wiki/Test_fixture

assertThat(mailFrom[0].toString()).isEqualTo(mailConfig.getFrom());
assertThat(mailSubject).isEqualTo(mailConfig.getSubject());
assertThat(destinataires).containsExactlyInAnyOrder(InternetAddress.parse(mailConfig.getTo()));   }

2.Ldap: Spring Embedded Ldap + UnboundID LDAP SDK allow to instantiate a LDAP server supporting LDAPv3 protocol with the ability to initialize inputs and extend schemas via ldif files.

SpringBoot configuration:

#ldapspring.ldap.embedded.base-dn=dc=brgm,dc=frspring.ldap.embedded.credential.username=uid=adminspring.ldap.embedded.credential.password=secretspring.ldap.embedded.ldif=classpath:data.ldif
spring.ldap.embedded.port=12345spring.ldap.embedded.validation.enabled=truespring.ldap.embedded.validation.schema=classpath:schema-custom.ldif

5.6 Security tests

We plan to have dedicated time to present and train the different team to use Owasp ZAP[15]. Each team would then report on the scan results, leading to potential security improvement.

5.7 FAIR assessment

We want to use WP3 recommendation tool: https://inspire.ec.europa.eu/validator/
We might also use: https://github.com/opengeospatial/teamengine

5.8 Module improvements

We have seen that the GIP-P project has produced an important effort in development and has made significant improvement in regards to the user expectations.

5.9 Conclusion

We suggested from the very start to have a pragmatic approach and focus on the easiest actions because this deliverable has been written halfway through software delivery plan. This pragmatic approach translates into focusing on the user review and the coherence and global understanding of the EGDI platform.
Following that mind-set, and throughout this deliverable, we have been able to achieve progress on multiple layers: a) the global understanding of the system, b) the solicitation of the system, c) the user satisfaction so far, d) define actions on the testing framework and e) on the module improvements
This stage in the testing framework is presented by Figure 6. It gives the rational for the next testing activities.

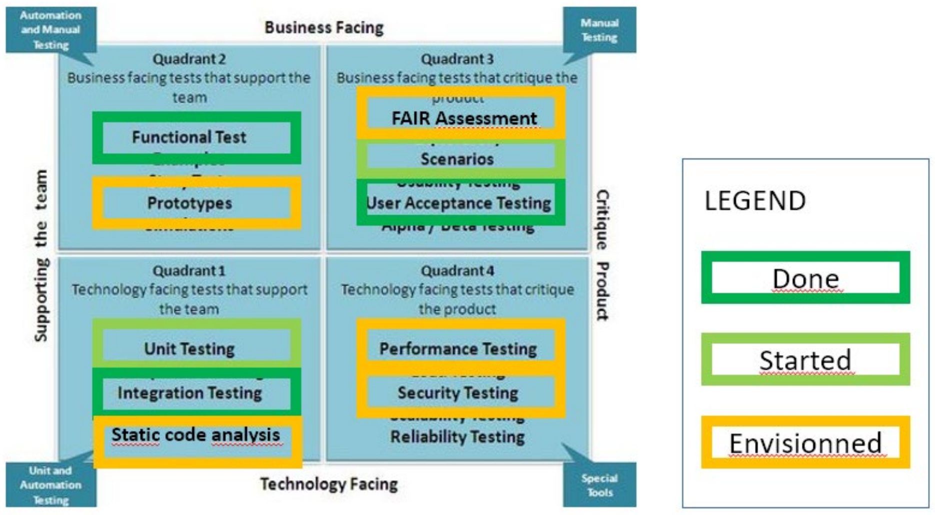


Figure 49: Testing quadrant status

We suggest to have a review of this document that would demonstrate this improvement of the status of the EGDI platform in regards to the testing strategy.

[15] https://owasp.org/www-project-zap/

6.ANNEX: EGDI DOCUMENT REPOSITORY FUNCTIONNALITIES

When you go to the URL: https://www.geo-zs.si/db/egdi-search/ you get the home page of the EGDI Repository Search platform.

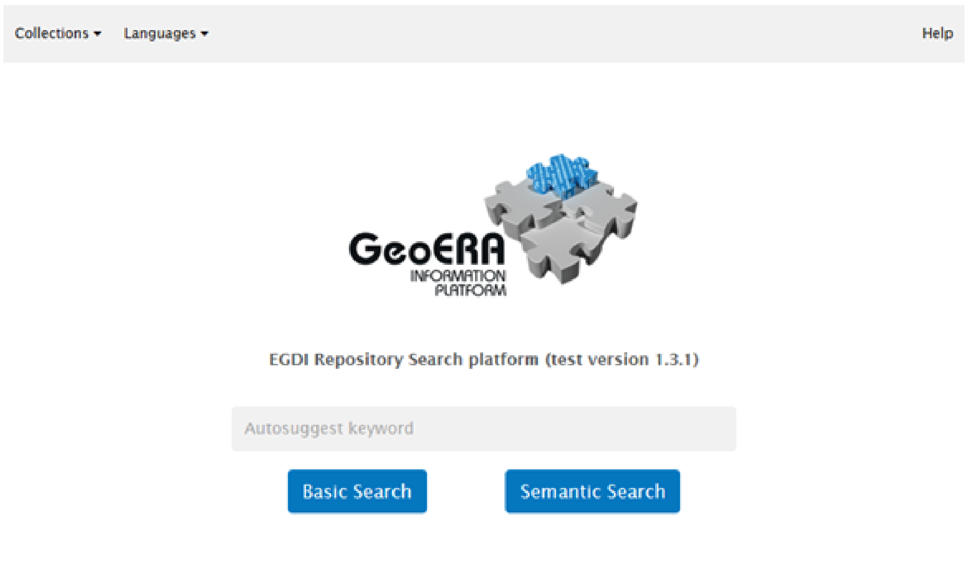
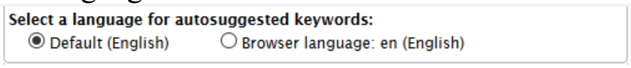

Collections
Languages
Help
GeoERA INFORMATION PLATFORM
EGDI Repository Search platform (test version 1.3.1)
Autosuggest keyword
Basic Search
Semantic Search

Figure 50: The home page

Settings menu

In the navigation bar of the application, the user can optionally set the following options in the settings menu:

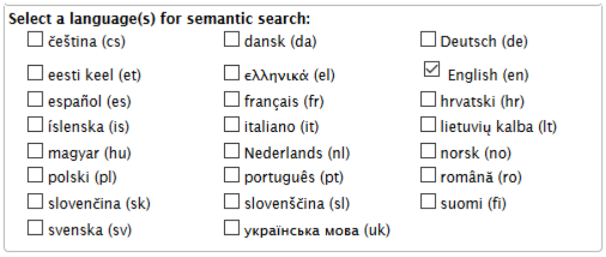
- Language of the autosuggested keywords list
    - English (default) - predefined
    - Current language of the user's browser



Select a language for autosuggested keywords:

[✓ Default (English)] Browser language: en (English)

Figure 51: Language(s) for the semantic search



Select a language(s) for semantic search:

☐ čeština (cs)	☐ dansk (da)	☐ Deutsch (de)
☐ eesti keel (et)	☐ ελληνικά (el)	☑ English (en)
☐ español (es)	☐ français (fr)	☐ hrvatski (hr)
☐ íslenska (is)	☐ italiano (it)	☐ lietuvių kalba (lt)
☐ magyar (hu)	☐ Nederlands (nl)	☐ norsk (no)
☐ polski (pl)	☐ português (pt)	☐ română (ro)
☐ slovenčina (sk)	☐ slovenščina (sl)	☐ suomi (fi)
☐ svenska (sv)	☐ українська мова (uk)	

Figure 52

- Collections from which to get the search results
    - egdi-images
    - egdi-documents
    - egdi-data

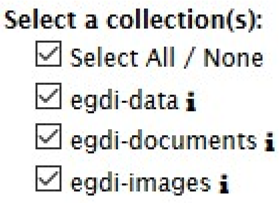
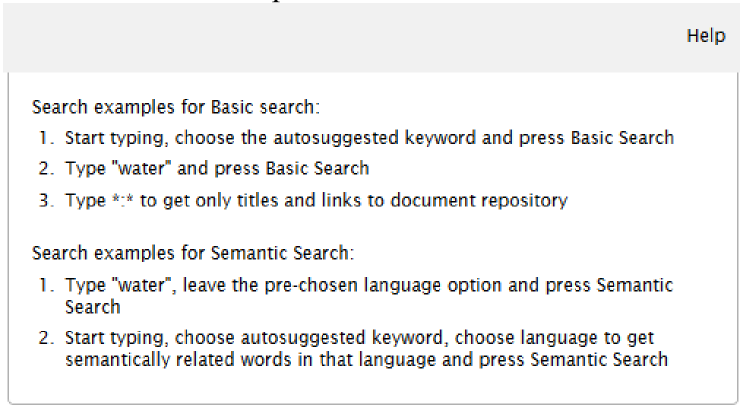

Select a collection(s):
- ☑ Select All / None
- ☑ egdi-data
- ☑ egdi-documents
- ☑ egdi-images

Figure 53

- Type of search (to be implemented)
  - basic search
  - semantic search
  - advanced search (to be implemented)
- Spatial search (to be implemented)
  - on/off and
  - contains/intersects buttons

In the navigation there is also the help information available on mouse hover.


Help

Search examples for Basic search:
1. Start typing, choose the autosuggested keyword and press Basic Search
2. Type "water" and press Basic Search
3. Type *:* to get only titles and links to document repository

Search examples for Semantic Search:
1. Type "water", leave the pre-chosen language option and press Semantic Search
2. Start typing, choose autosuggested keyword, choose language to get semantically related words in that language and press Semantic Search

Figure 54: The tips popup window on mouse hover

<u>Input search box</u>
The user then starts entering the characters of the search term in the input search box. As he starts typing, the autosuggested keywords box pops up with suggestions.

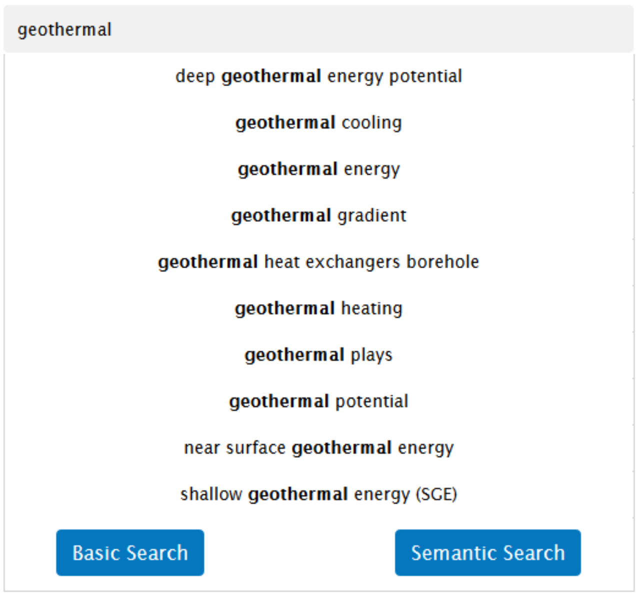


Figure 55: The list of autosuggested keywords

The user can then click on the specific word from the autosuggested keywords list or confirms the word with [Enter]. We plan to enable searching by multiple keywords (see Figure 8)

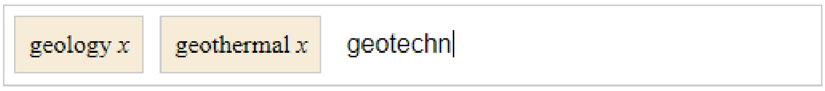


Figure 56: (this type of search is not implemented yet as in version 1.3.2)

Then the user performs the search with a click on the Search button. Based on the search type, additional HTTP GET requests are executed under the hood.

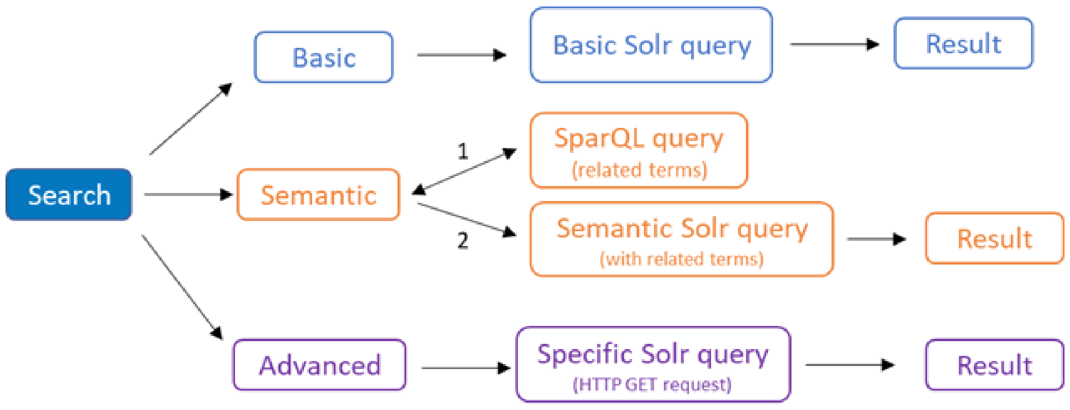
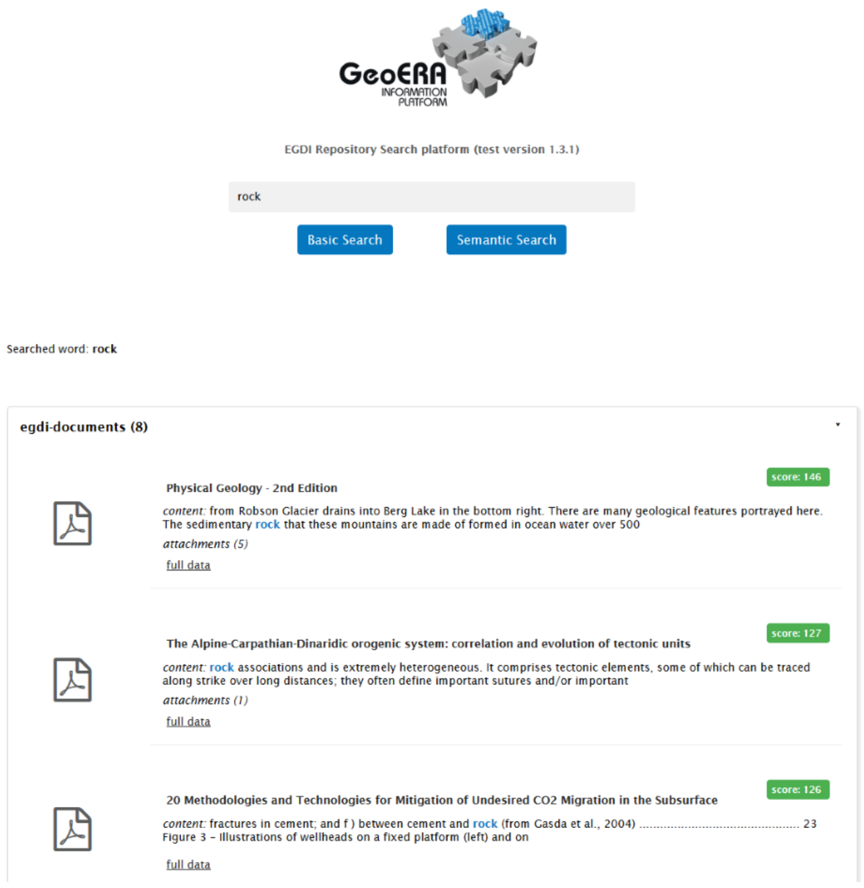


Figure 57: Search type procedure

<u>Example of a Basic search:</u>

- User types the searched word(s) and presses Search button.
- The searched word(s) are displayed and highlighted in the results which are grouped and displayed to the user.


EGDI Repository Search platform (test version 1.3.1)

rock

Searched word: **rock**

egdi-documents (8)

score: 146
Physical Geology - 2nd Edition
content: from Robson Glacier drains into Berg Lake in the bottom right. There are many geological features portrayed here. The sedimentary rock that these mountains are made of formed in ocean water over 500
attachments (5)
full data

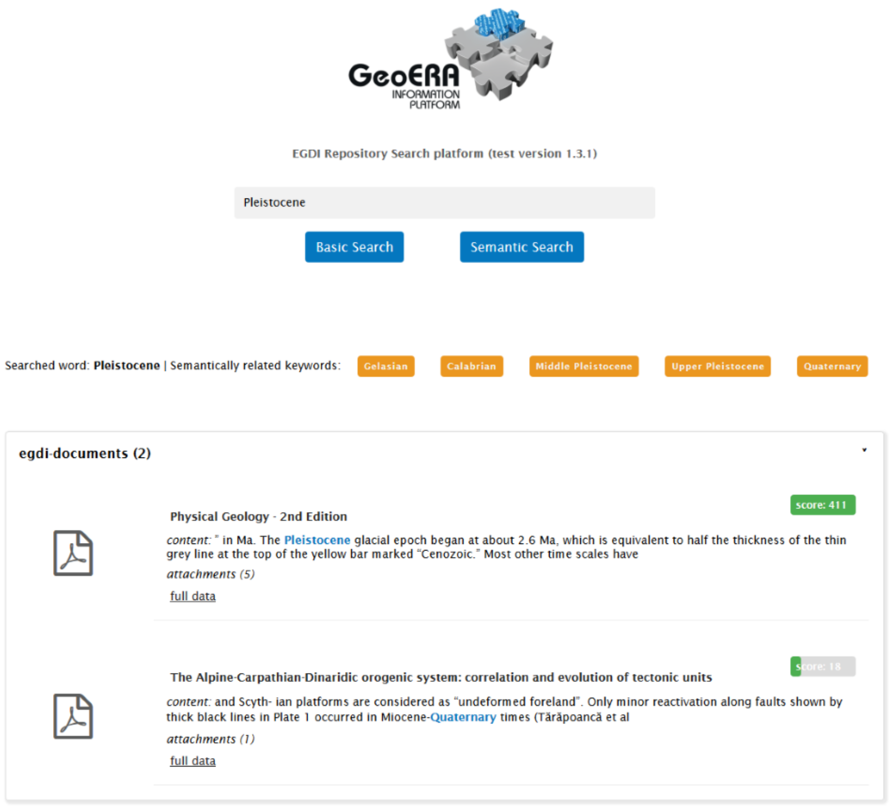
score: 127
The Alpine-Carpathian-Dinaridic orogenic system: correlation and evolution of tectonic units
content: rock associations and is extremely heterogeneous. It comprises tectonic elements, some of which can be traced along strike over long distances; they often define important sutures and/or important
attachments (1)
full data

score: 126
20 Methodologies and Technologies for Mitigation of Undesired CO2 Migration in the Subsurface
content: fractures in cement; and f) between cement and rock (from Gasda et al., 2004) .... 23 Figure 3 – Illustrations of wellheads on a fixed platform (left) and on
full data

Figure 58

<u>Example of a Semantic search:</u>

- User types the searched word(s) and presses Search button.
- The searched word and related terms for a specific language are displayed and highlighted in the results which are grouped and displayed to the user.



EGDI Repository Search platform (test version 1.3.1)

Pleistocene

Searched word: **Pleistocene** | Semantically related keywords: Gelasian Calabrian Middle Pleistocene Upper Pleistocene Quaternary

egdi-documents (2)

Physical Geology - 2nd Edition score: 411

content: " in Ma. The Pleistocene glacial epoch began at about 2.6 Ma, which is equivalent to half the thickness of the thin grey line at the top of the yellow bar marked "Cenozoic." Most other time scales have

attachments (5)

full data

The Alpine-Carpathian-Dinaridic orogenic system: correlation and evolution of tectonic units score: 18

content: and Scyth- ian platforms are considered as "undeformed foreland". Only minor reactivation along faults shown by thick black lines in Plate 1 occurred in Miocene-Quaternary times (Tărăpoancă et al

attachments (1)

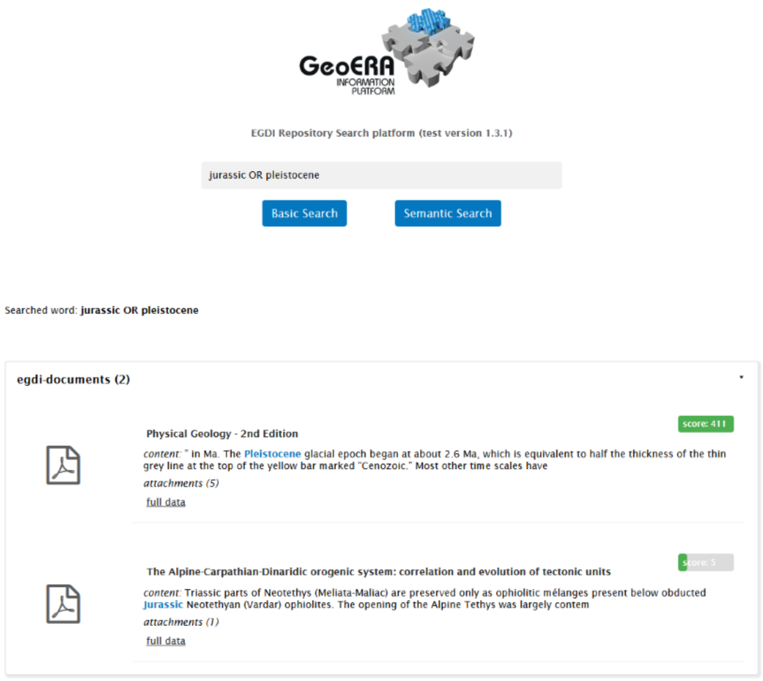
full data

Figure 59

<u>Example of Advanced search:</u>

4. User can type specific Filter query parameters, e.g.:
    a. Author: "Nina Rman" - search returns only the documents where one of the authors is "Nina Rman"

- Bottled AND mineral water - (boolean operator) search returns only the documents which contain both terms. Other boolean operators include OR, NOT.


EGDI Repository Search platform (test version 1.3.1)

jurassic OR pleistocene

Searched word: **jurassic OR pleistocene**

egdi-documents (2)

Physical Geology - 2nd Edition score: 411

content: " in Ma. The Pleistocene glacial epoch began at about 2.6 Ma, which is equivalent to half the thickness of the thin grey line at the top of the yellow bar marked "Cenozoic." Most other time scales have

attachments (5)

full data

The Alpine-Carpathian-Dinaridic orogenic system: correlation and evolution of tectonic units score: 5

content: Triassic parts of Neotethys (Meliata-Maliac) are preserved only as ophiolitic mélanges present below obducted Jurassic Neotethyan (Vardar) ophiolites. The opening of the Alpine Tethys was largely contem

attachments (1)

full data

Figure 60

Spatial search selector

The user interface will also implement the geographically aware type of search in combination with the three types of searches mentioned above. The user will be able to choose the area of interest in the web map and then cross examine this area with conjunction to input field values.

The results (as in this version) are grouped by collections and descended based on the Solr matching score. The result from each matching document that gets rendered to the page consist of:
Highlighted text of the matching part of the document from the specific metadata field or content with the matched keywords
Document’s title and URL to access it
Image thumbnail (in case of the image collection) or icon describing the type of the document
Full metadata link - link to the full record’s data that pops up in a separate browser tab
Attachments – document's attachments that link to the data (images, pdf documents…) related to the main document. For image attachments, the small thumbnail preview of the attached files is also displayed. Attachments also link to the full-size image or, in case of documents, pdf document.

7.REFERENCES

D 2.2.1: First report describing the requirements to the Information Platform by the Geo-energy, Groundwater and Raw Materials themes. January 2019. https://geoera.eu/wp-content/uploads/2019/01/D2.2.1-Requirements-to-the-Information-Platform.pdf.

D2.2.2: A second report refining the requirements after feedback exchanges related to the prototypes of the EGDI database and the display interface. January 2020. https://geoera.eu/wp-content/uploads/2020/01/D2.2.2-Refinements-of-requirements.pdf; https://geoera.eu/wp-content/uploads/2020/01/D2.2.2-Appendix-A.pdf.

D 2.3.1: First report mapping and describing the needed extensions to EGDI directly related to the task 2.2. March 2019.

D 2.1.1: First report highlighting the potential synergies and overlaps between the projects in terms of geoinformation. June 2019. https://geoera.eu/wp-content/uploads/2019/07/D2.1.1-Potential-synergies-and-overlaps.pdf.

D4.2: Keyword Thesaurus. October 2019. https://geoera.eu/wp-content/uploads/2019/11/D4.2-GeoERA-Keyword-Thesaurus.pdf.

D4.3: GeoERA project vocabulary. October 2019. https://geoera.eu/wp-content/uploads/2019/11/D4.3-GeoERA-Project-Vocabularies.pdf.

D7.1: Working version Metadatabase. December 2019. https://geoera.eu/wp-content/uploads/2019/12/D7.1-Working-version-Metadatabase.pdf

Kramolišová, P., Kondrová, L., Moravcová, O., and Kafka, Š.: Cookbook for creating metadata records using the EGDI Metadata catalogue (MIcKA, version 6.0). April 2020.

Mintell4EU D5.3.1 Specification of steps needed for the integration of the E-MYB in the M4EU DB.