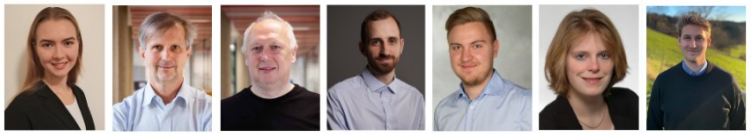
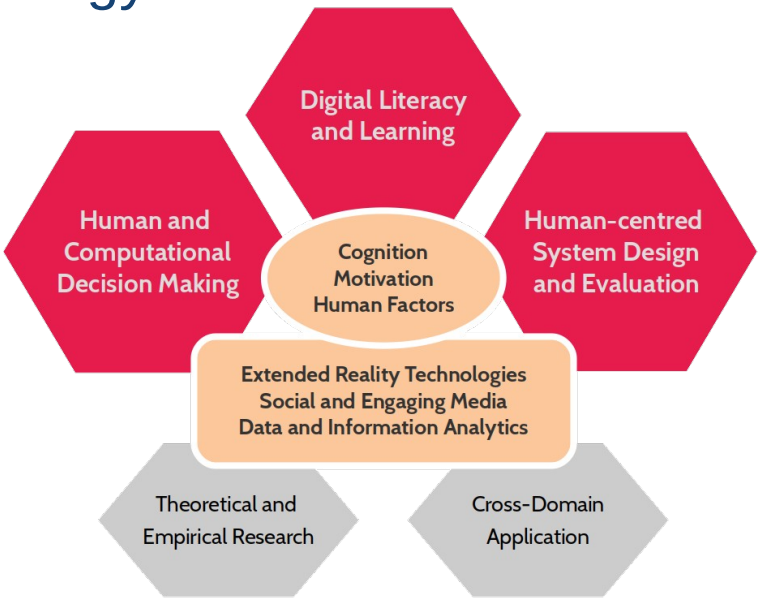

Funded by
the European Union
SUPPORTED BY
NGI
Hackathon
Introduction
Part 1
Alexander Nussbaumer, Sebastian Gürtl,
Christian Gütl
Graz, Austria
24 May 2024

Organisers

- CoDiS Lab @ ISDS, TUGraz
  - Interdisciplinary research overlapping computer science and cognitive psychology
  - 7 Staff members
  - Phd, master and bachelor students

http://isds.tugraz.at/codis/





Agenda

09:00 – 09:15	Welcome
09:15 – 09:45	**Introduction (*)**
09:45 – 10:00	Group formation and coffee
10:00 – 11:00	Working session 1: Idea generation
11:00 – 11:15	**Presentation of ideas (*)**
11:15 – 13:00	Working session 2
13:00 – 14:00	Lunch – Mensa Mercato
14:00 – 17:00	Working session 3
17:00 – 18:00	**Presentation of results, voting and prices (*)**
18:00 – 19:00	Get-together & beer; Lange Nach der Forschung

Organisation

- Communication
  - channels on Discord.com (general and one per team)
  - online broadcasting via Jitsi
- Collaboration in teams (2 – 4 people)
  - Online team: Jitsi breakout room or Discord voice channel
  - Self-organisation (e.g. GitLab)
- Information
  - Cloud folder with documents (presentations, developer guide, etc.)
  - Sub-folders for teams (project submission)

Organisation

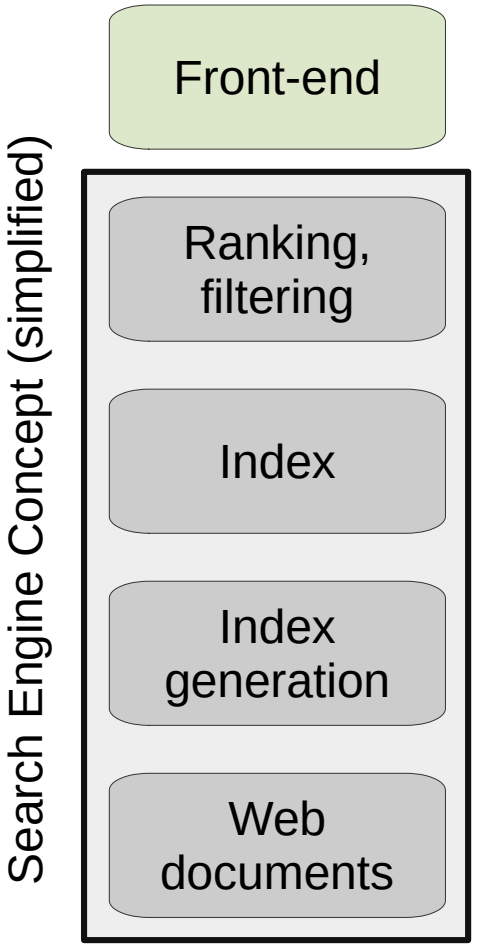
- Coffee in kitchen
- Cookies, mineral water, juice
- Lunch in Mensa Mercato
- Free beer after event
- Lange Nacht der Forschung

Hackathon Goal and Concept

- Introduction to OpenWebSearch.eu and MOSAIC
  - Presentation, manual, explanation and discussion
- Enrich or build an alternative web search solution
  - elaboration of a concept and use case
  - development or improvement of a MOSAIC component
- Result presentation
  - Discussion of results and voting

Background and Motivation

- Web search is dominated by a few big players
  - Google, Bing, Yandex, Baidu
- Few alternatives available
  - Brave, Qwant,
  - Meta search engines, e.g. Ecosia, DuckDuckGo
- Goal is the creation of an Open Web Index
  - can be used for an own search application



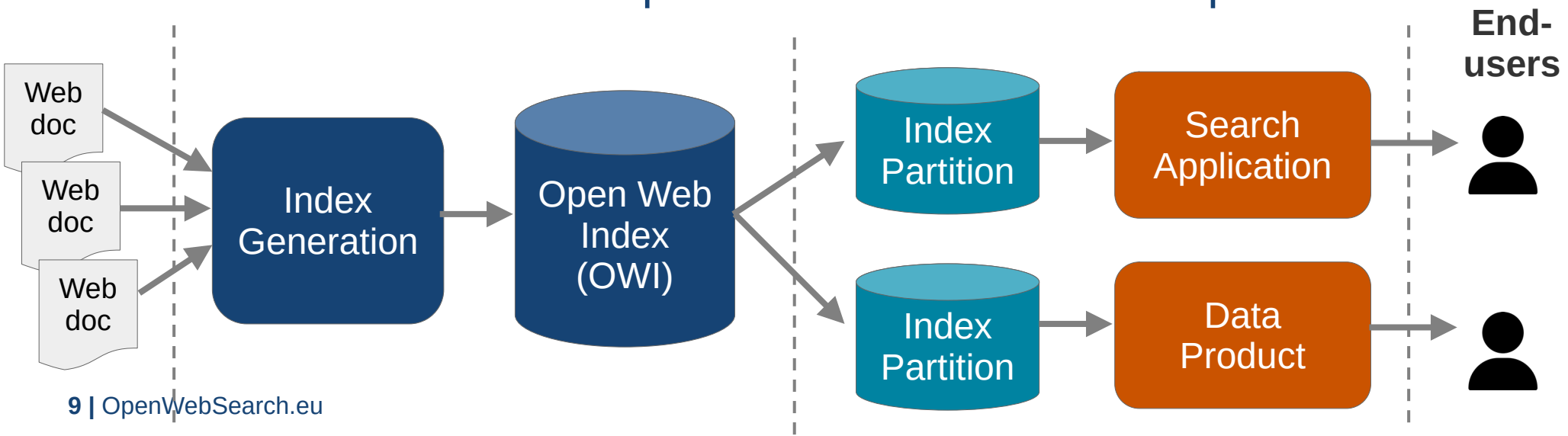
OpenWebSearch.eu

- Horizon Europe Research Project
  - 14 partners (research institutions and data centers)
- Objectives (simplified)
  - Development of an Open Web Index and related software
  - Development of relevant search engine verticals and ecosystem of vertical search engines

Open Web Index Status March 2024

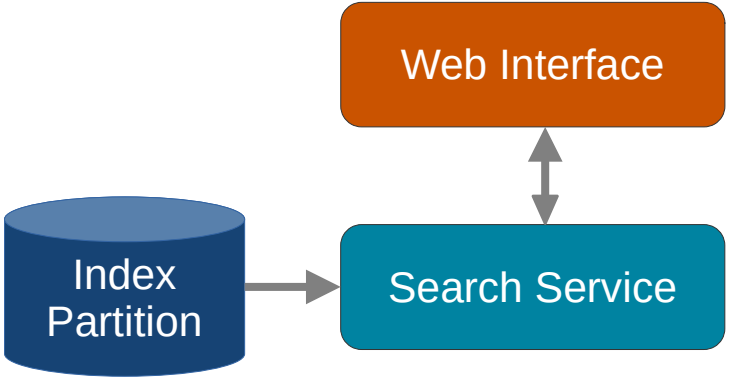
Vertical Search Engines in OWS.eu

- Vertical Search Engines serve special purposes, topics, or domains, e.g. product search
- Search applications are a key concept in OpenWebSearch.eu and are based on the Open Web Index and index partitions



MOSAIC

- **Mo**dular **S**earch **A**pplication based on **I**ndex Fra**c**tions
- Generic implementation of an OWS.eu vertical search engine
  - Demonstration of the concept of an OWS.eu vertical engine
  - Out-of-the-box search engine
  - Toolbox for an own search application



MOSAIC Front-end

Open WebSearch .eu

1 **Search term:** cern Search Index Info

2 **Geo Filter:** West: 1.8 East: 17.0 North: 55.6 South: 40.2

3 **Index:**
- ○ default / all
- ◉ Demo SimpleWiki
- ○ Demo Graz Universities
- ○ DLR Prototype

4 **Language:**
- ○ default / all
- ◉ English
- ○ German

Limit:
- ◉ default / 20
- ○ 10 items
- ○ 50 items
- ○ 1,000,000

5 **Keyword:**

6 **Search URL:** https://qnode.eu/ows/mosaic/service/search?q=cern&index=demo-simplewiki&lang=eng&west=1.8&east=17.0&north=55.6&south=40.2

7 **Wikipedia: World Wide Web**

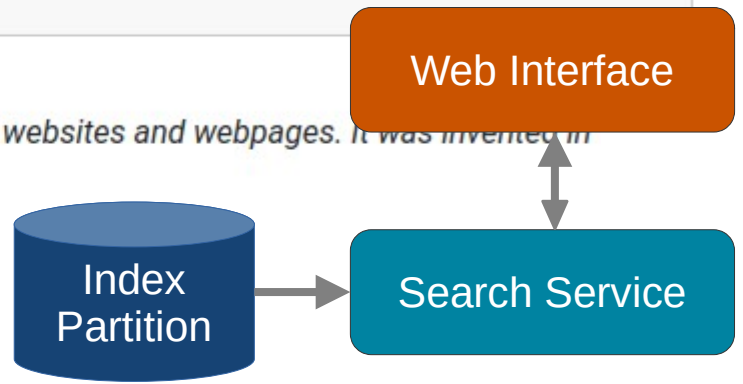
The World Wide Web ("WWW" or "The Web") is the part of the Internet that contains websites and webpages. It was invented in 1989 by Tim Berners-Lee at CERN, Geneva, Switzerland.

8 Metadata: *language:eng, word count:36, index date:NaN-NaN-NaN NaN:NaN*

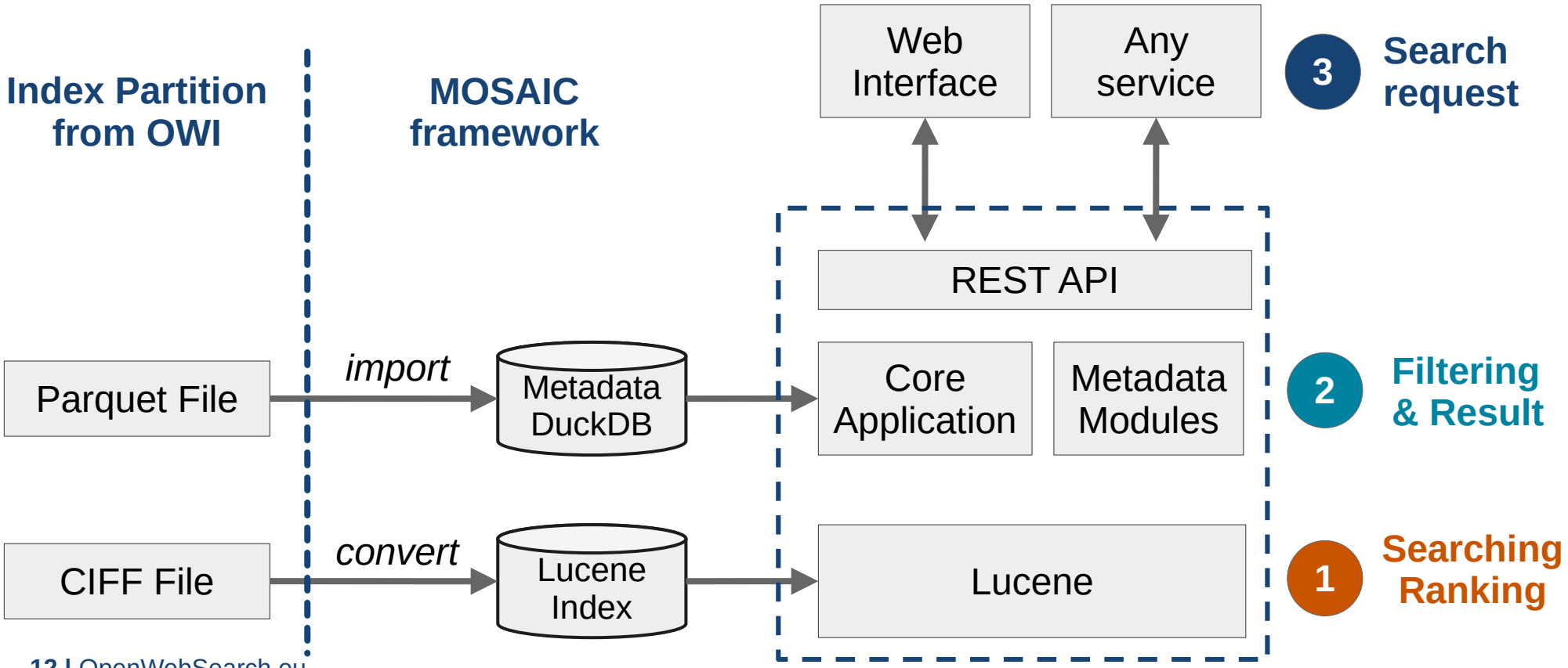
Locations: Geneva • Switzerland •

Keywords:

https://simple.wikipedia.org/wiki/World_Wide_Web



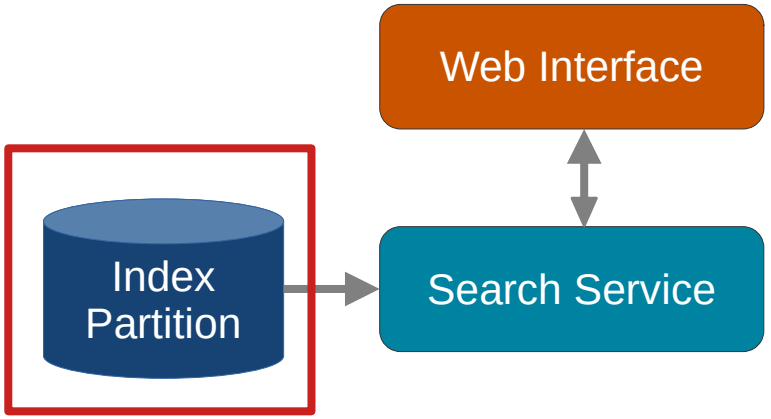
MOSAIC Concept



Development Possibilities

1) Create your own index

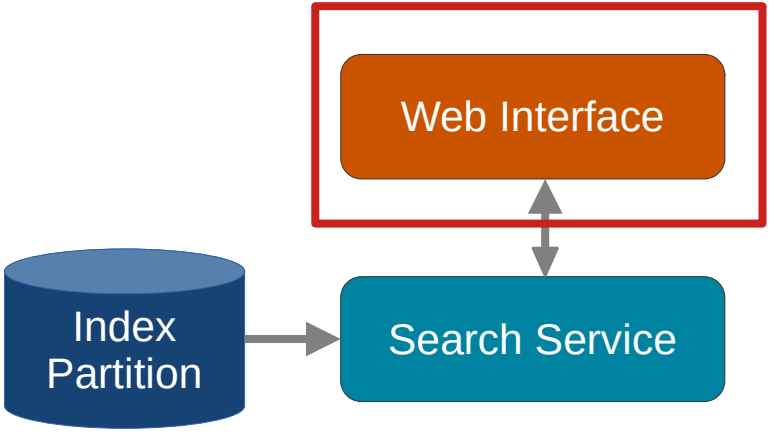
- Download an existing one
- Create an index with a list of URLs



Development Possibilities

2) Create or improve a front-end

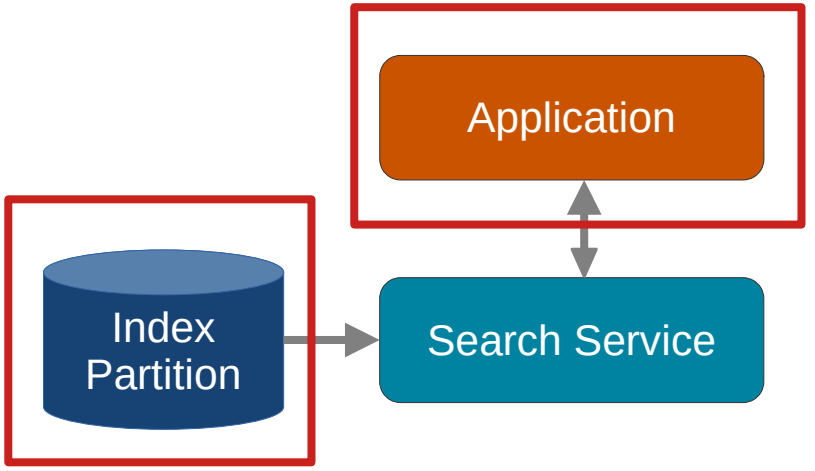
- user interface design
- selecting filter options
- display result attributes



Development Possibilities

3) Create an application that uses MOSAIC as service

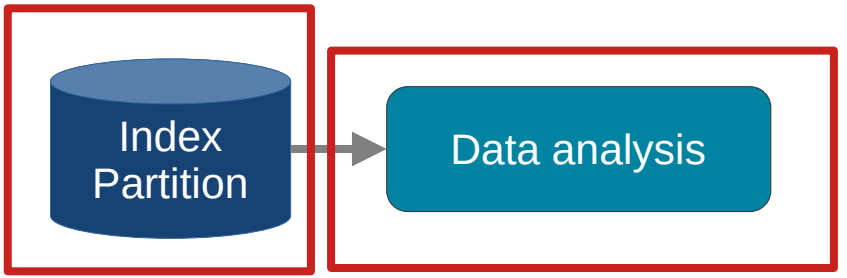
- develop a use case and application idea
- select or create an index
- use the REST API of MOSAIC
- implement the application



Development Possibilities

4) Undertake web data analysis

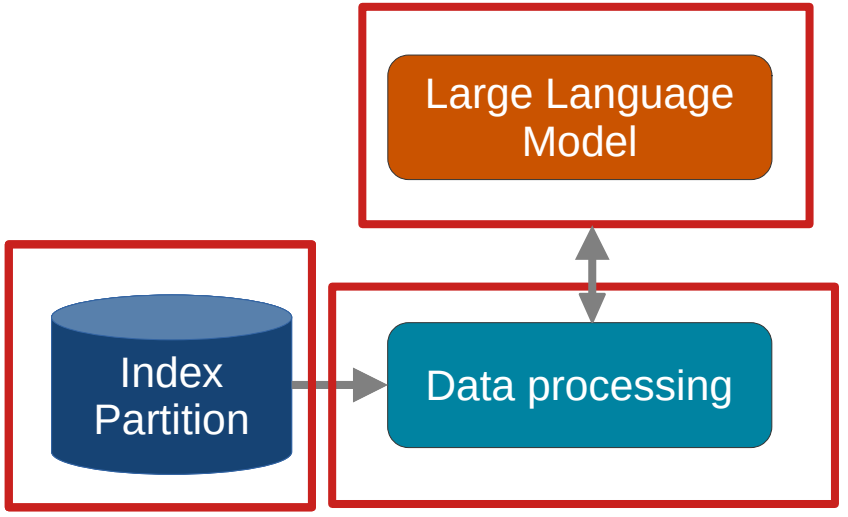
- develop a use case and analysis goal
- select or create an index
- implement an analysis procedure



Development Possibilities

5) Feed a Large Language Model

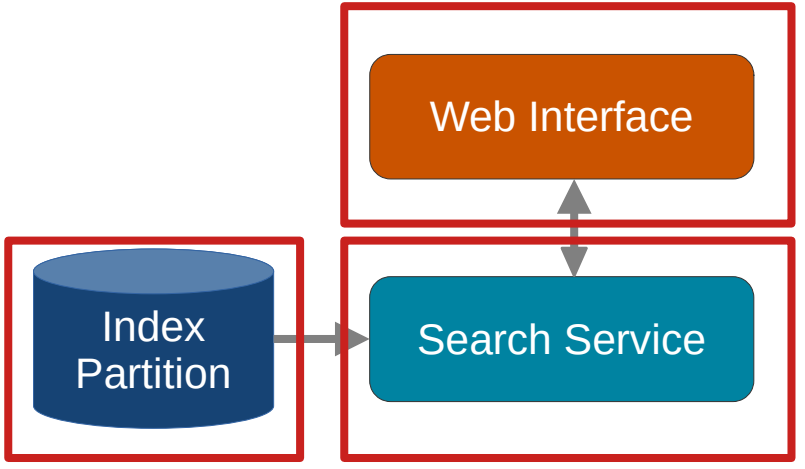
- develop a use case and analysis goal
- select or create an index
- select and feed the LLM



Development Possibilities

6) Create a new module

- use Parquet file with additional metadata columns
  or create your own additional metadata on your own
- create a module that provides filtering and returning results
- adapt the front end

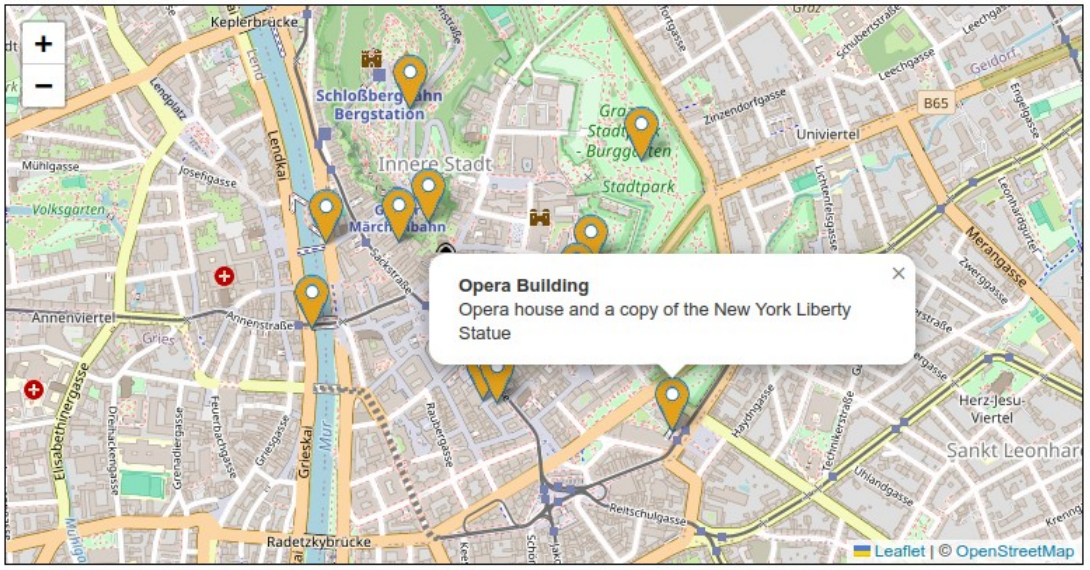


Application Example

- Sightseeing spots
- Clicking on a spots on a map triggers a search and displays result
- Sightseeing index has been created for this application

https://qnode.eu/ows-spots/

Sightseeing Graz



Search result for term: Opera Building

Best attractions in Graz

The cafe, opened in a hotel with the same name, can be called a full-fledged Austrian attraction, because it is here that they serve the real Sacher cake, cooked according to the original ancient recipe. Keep in mind that there are always a lot of vi

https://www.tripzaza.com/en/destinations/top-attractions-in-graz

Opera House Graz

Something fascinating about Graz - the interplay of modernism and tradition - is illustrated by the sculpture "light sword" next to the opera house. It was originally made for the festival "steirischer herbst" in 1992. To celebrate the 500th annivers

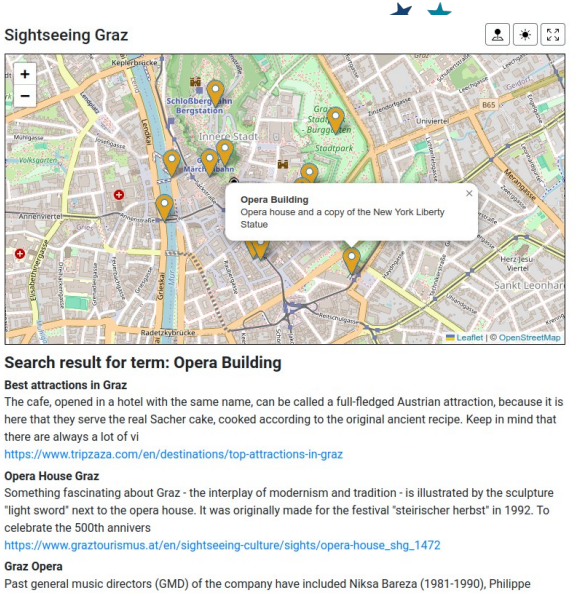
https://www.graztourismus.at/en/sightseeing-culture/sights/opera-house_shg_1472

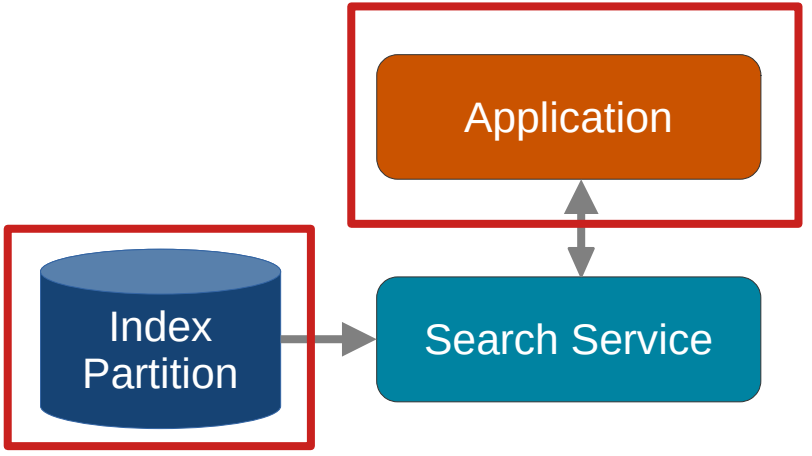
Graz Opera

Past general music directors (GMD) of the company have included Niksa Bareza (1981-1990), Philippe

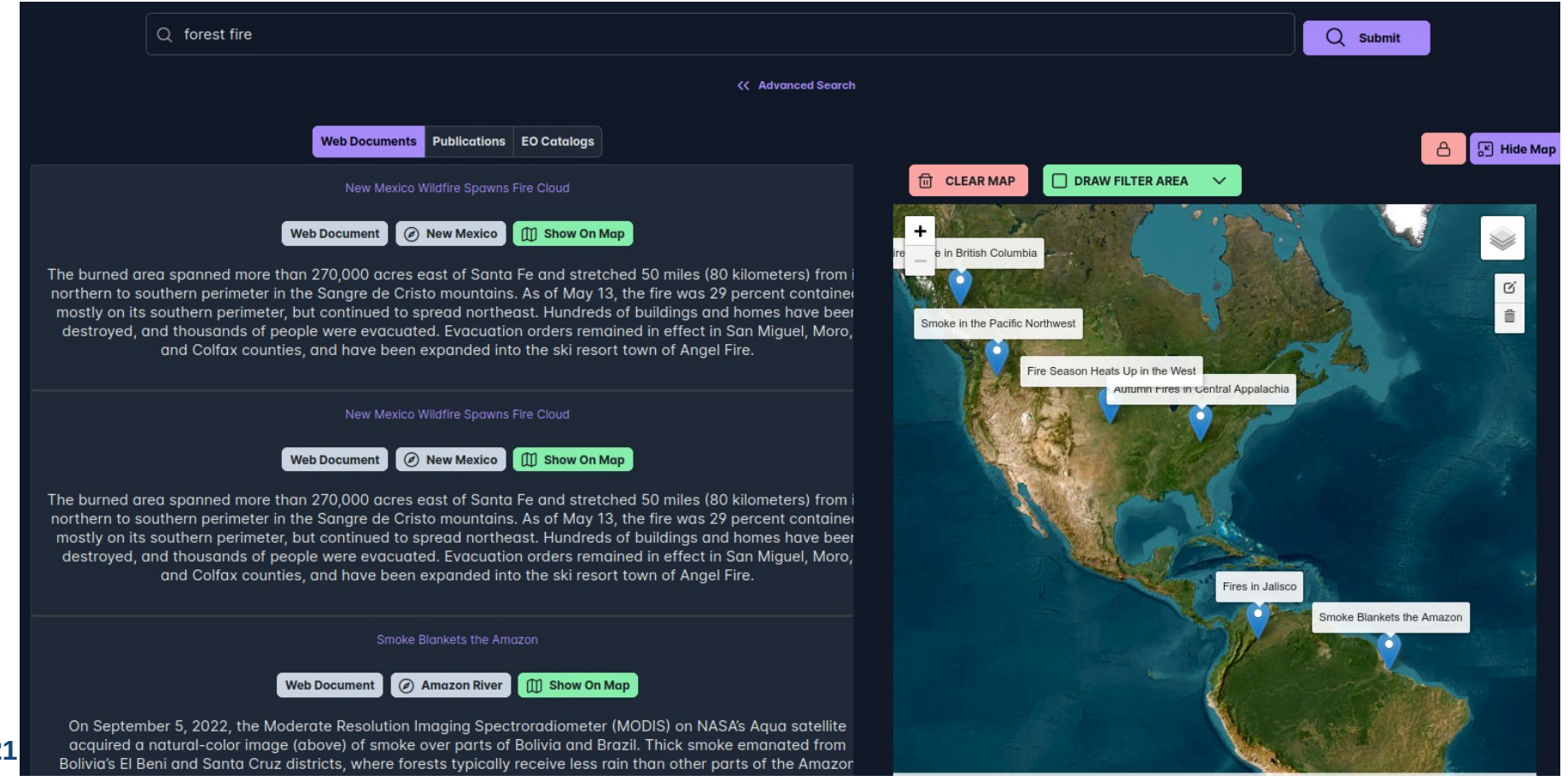
Application Example

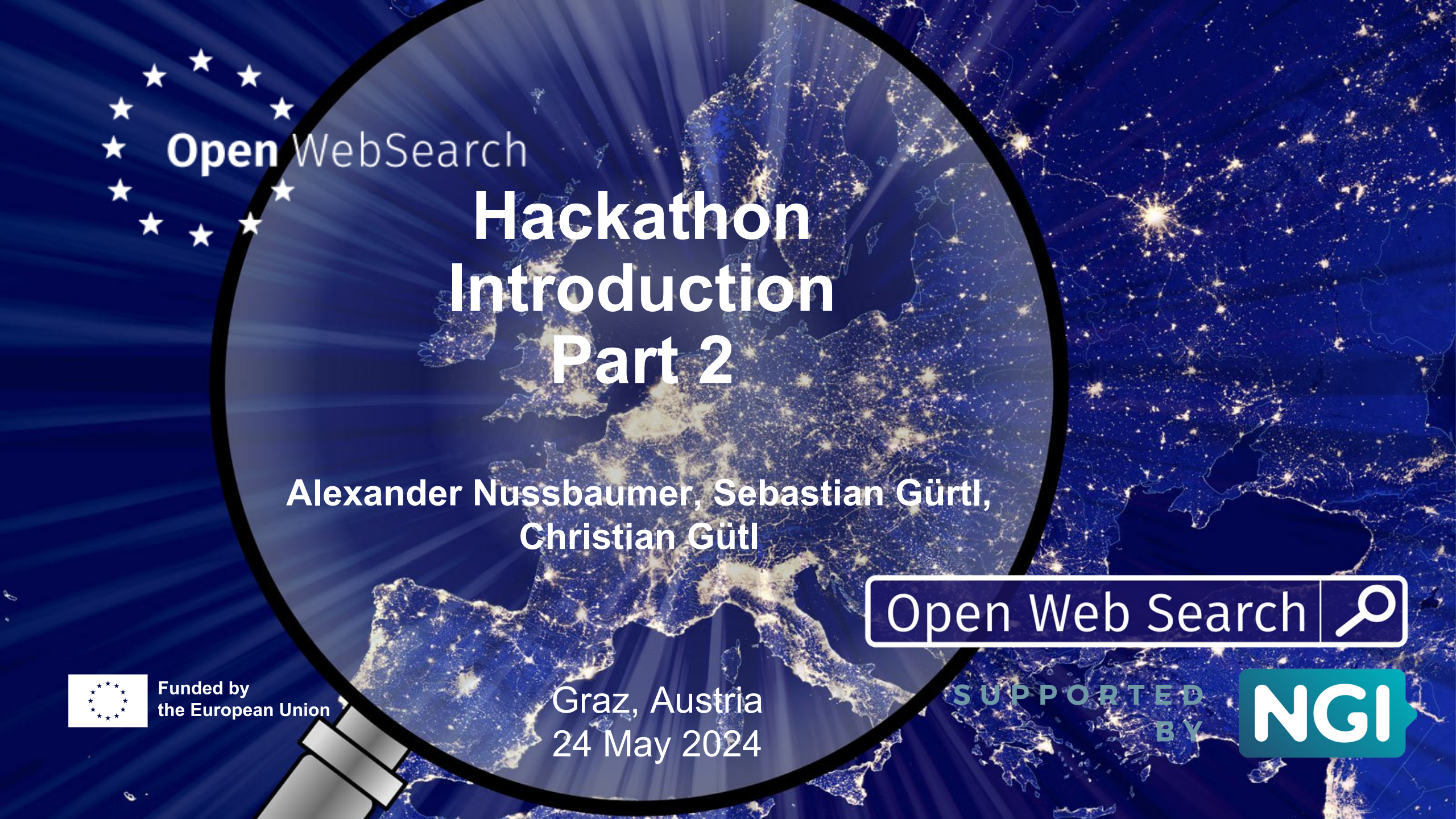
- Index:
  - generated from 50 web pages
- Search service
  - used out-of-the-box
- Web interface
  - Created with Leaflet and added search queries in click event





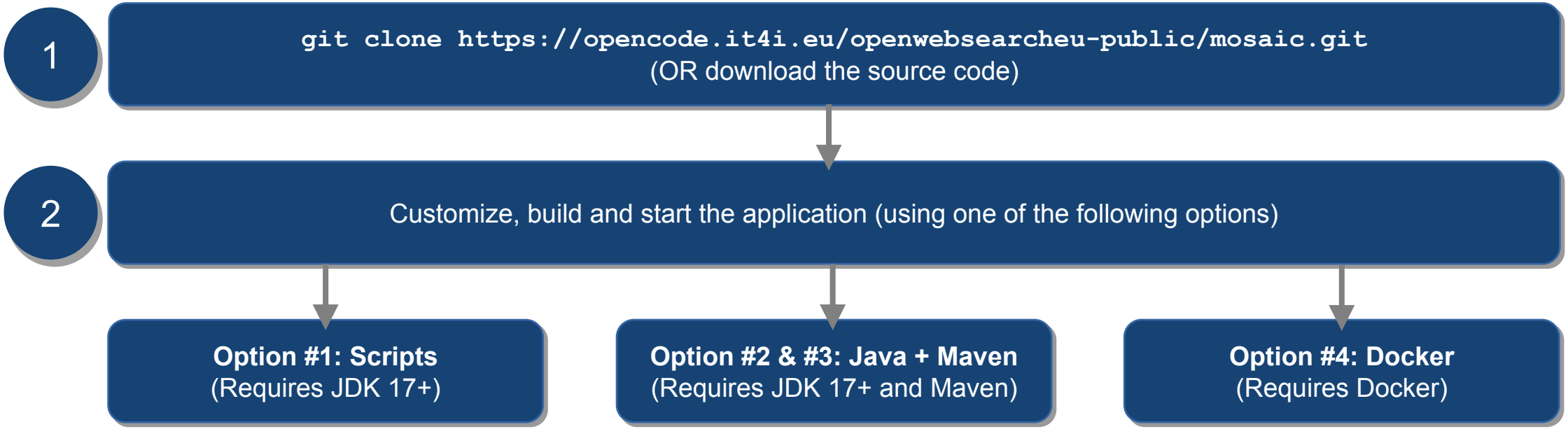
Application Example: Science Search




Open WebSearch
Hackathon Introduction Part 2
Alexander Nussbaumer, Sebastian Gürtl, Christian Gütl
Open Web Search
Funded by the European Union
Graz, Austria
24 May 2024
SUPPORTED BY
NGI

Next Steps Towards Creating Your Own Web Search Solution

➜ Find the source code of the MOSAIC on GitLab:
https://opencode.it4i.eu/openwebsearcheu-public/mosaic



MOSAIC Installation Guide
Option #1: Scripts

➜ Prerequisites: Java (JDK 17+)

```
# change the directory after cloning
cd scripts
# import indexes (optional), clean the project and create a packaged JAR
./build.sh
# run the executable
./start.sh "[OPTION]" [API_PORT]
```

➜ Example: **`./start.sh`** (both **`OPTION`** and **`API_PORT`** are optional)

➜ Batch files as alternative for Windows

MOSAIC Installation Guide
Option #2: Java + Maven

➜ Prerequisites: Java (JDK 17+) + Maven (3.8.2+)

```
# change the directory
cd search-service
# clean the project and compile the source code
mvn clean compile
# build the executable
mvn package
# run the executable
java [-Dquarkus.http.port=<API_PORT>] -jar core/target/service.jar
[OPTION]
```

➜ Example: **`java -jar core/target/service.jar`**

MOSAIC Installation Guide
Option #3: Dev Mode

➜ Prerequisites: Java (JDK 17+) + Maven (3.8.2+)

```
# change the directory
cd search-service
# run MOSAIC in dev mode
mvn quarkus:dev [-Dquarkus.http.port=<API_PORT>] [-Dquarkus:args="OPTION"]
```

➜ Example: **`mvn quarkus:dev`**

➜ Enables Quarkus Live Coding

MOSAIC Installation Guide
Option #4: Docker

➜ Prerequisites: Docker

```
# create an image
cd scripts docker build -t mosaic .
# start a container
docker run --rm -p 8008:8008 mosaic [OPTION]
```

➜ Example: `docker run --rm -p 8008:8008 mosaic`

➜ Consider port binding of Docker if you want to change the port of MOSAIC

Index Partition: CIFF/Lucene and Parquet

- ➜ CIFF: Inverted Index Exchange Format
- ➜ Lucene: Inverted Index
- ➜ Parquet: Tabular Format

CIFF/Lucene

Term	List of (Document, Frequency)
Term 1	(docID A, 1) (docID B, 3) (docID C, 1)
Term 2	(docID D,2)
Term 3	(docID E, 4) (docID F, 2)
Term 4	(docID H, 2) (docID I, 1)

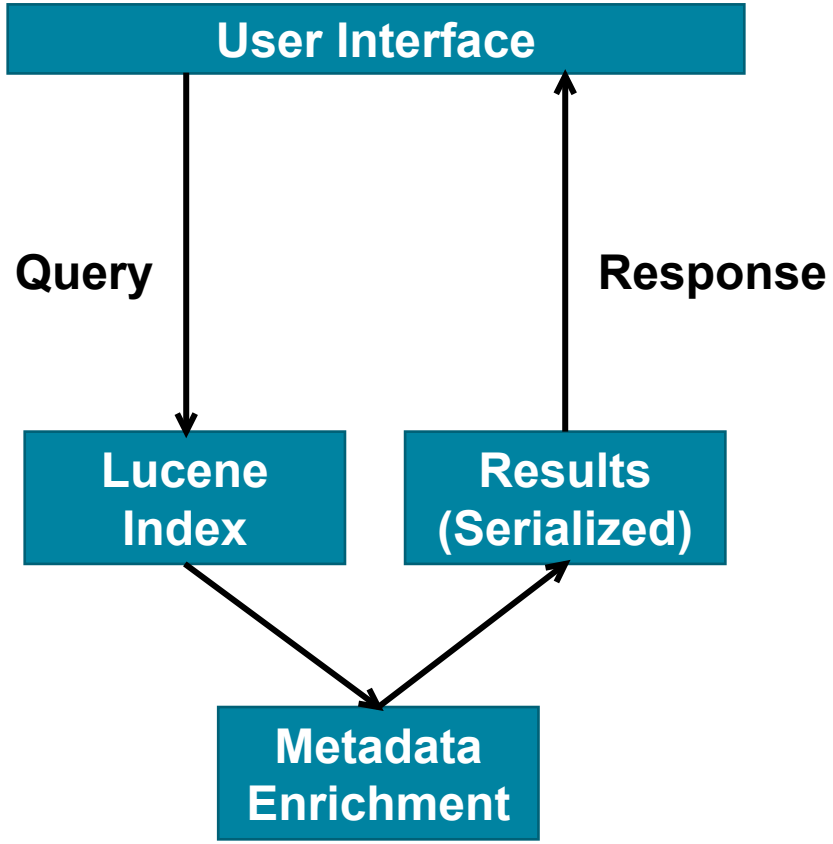
Parquet

Document	Lang	Full text	WARC date	...
docID A	eng	… text ...	2023-07-01	
docID B	deu	… text ...	2023-07-01	
docID C	deu	… text ...	2023-07-01	
docID D	eng	… text ...	2023-07-01	

Parquet File / Metadata Fields (Full List + Details)

General Metadata	URL Related Metadata
id	url
record_id	url_scheme
plain_text	url_path
title	url_params
language	url_query
domain_label	url_fragment
domain_labels	url_subdomain
warc_date	url_domain
warc_ip	url_suffix

Metadata Enrichment



Lucene DocID = Parquet DocID

General Metadata	URL Related Metadata
id	url
record_id	url_scheme
plain_text	url_path
title	url_params
language	url_query
domain_label	url_fragment
domain_labels	url_subdomain
warc_date	url_domain
warc_ip	url_suffix

REST API

End point: `host:port/search?q=...`

REST Query Parameters:

Parameter	Description
`q`	Search term(s)
`index`	Lucene index to be searched in
`lang`	Restrict searches to pages in language
`ranking`	Specify order of search result
`limit`	Set maximum number of returned results

Example:
https://qnode.eu/ows/mosaic/service/search?q=austria

Defined per module:

- ➜ **Core**: Basic search and filtering
- ➜ **Geo**: Geo-information (i.e., coordinates, location names)
- ➜ **Keywords**: Extracted keywords from full plain text
- ➜ Feel free to add your own modules

REST API

End point: `host:port/search?q=...`

Response (JSON):

```
{
  "results": [
    {
      "simplewiki": [
        {
          "id": "cfe49c84-2244-430e-83e3-fe3f30aae21e",
          "url":"https://simple.wikipedia.org/wiki/Anthem_of_Europe",
          "title": "Wikipedia: Anthem of Europe",
          "textSnippet":"https://simple.wikipedia.org/wiki/Anthem_of_Europe",
          "language": "spa",
          "warcDate": "2024-01-15T22:19:46Z",
          "wordCount": 11
        },
        ...
      ],
    },
    {
       "unis-graz": [
        { ... },
      ],
      ...
   },
   ...
  ]
}
```

REST API

End point: `host:port/searchxml?q=...`

Response (OpenSearch / XML):

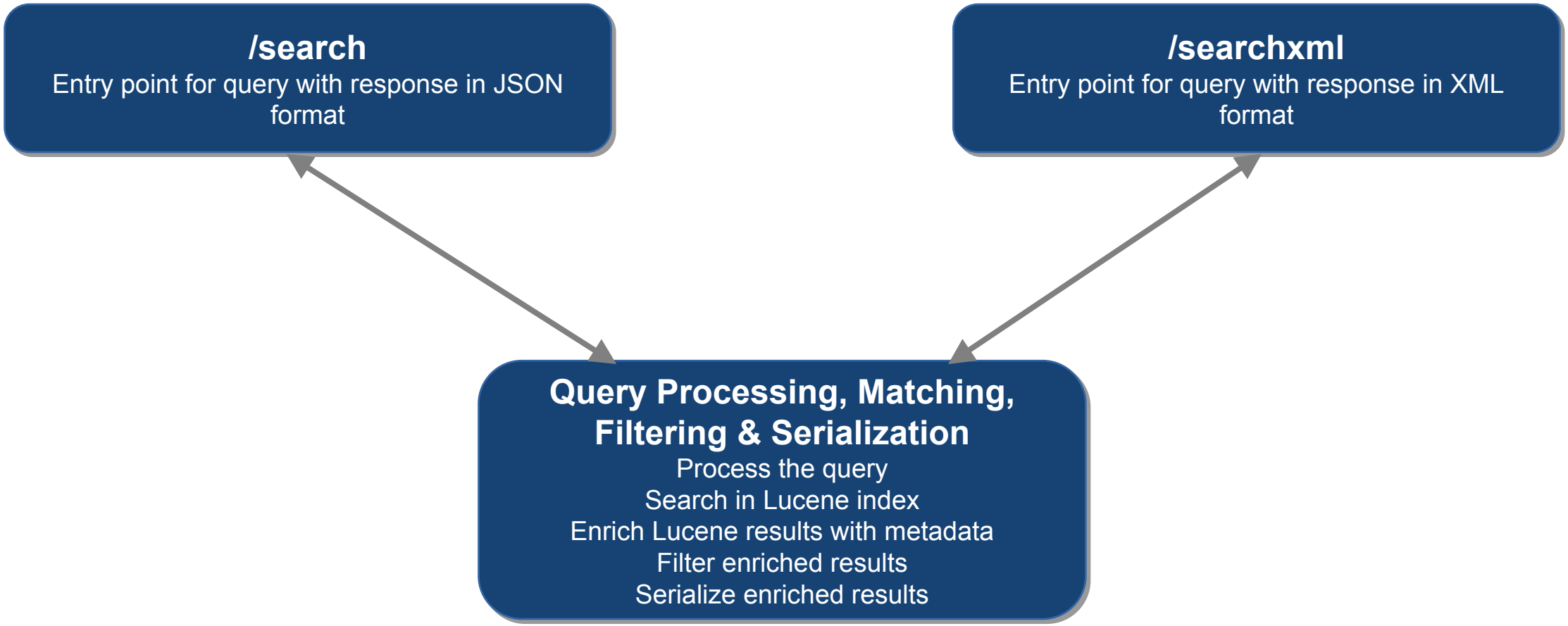
```
<feed xmlns="http://www.w3.org/2005/Atom" xmlns:opensearch="http://a9.com/-/spec/opensearch/1.1/">
  <title>MOSAIC Search: {searchTerms}</title>
  <description>Search results for "{searchTerms}" at MOSAIC Search Service</description>
  <author>
    <name>OpenWebSearch.eu</name>
  </author>
  <opensearch:totalResults>1121</opensearch:totalResults>
  <opensearch:startIndex>1</opensearch:startIndex>
  <opensearch:itemsPerPage>20</opensearch:itemsPerPage>
  <opensearch:Query role="request" searchTerms="{searchTerms}"
                                   startPage="1"/>
  <link rel="alternate" href="{baseUrl}/search?q={searchTerms}&pw=1&limit=20" type="application/json"/>
  <link rel="self" href="{baseUrl}/searchxml?q={searchTerms}&pw=1&limit=20" type="application/atom+xml"/>
  <link rel="next" href="{baseUrl}/searchxml?q={searchTerms}&pw=2&limit=20" type="application/atom+xml"/>
  <link rel="last" href="{baseUrl}/searchxml?q={searchTerms}&pw=56&limit=20" type="application/atom+xml"/>
  <link rel="search" type="application/opensearchdescription+xml" href="{baseUrl}/opensearch.xml"/>
  <item>
    ...
  </item>
  <item>
    ...
  </item>
  ...
</feed>
```

REST API

Additonal endpoints:

- ➜ **`/index-info`**: Returns a list of present indexes, including the number of documents and available languages for each index
- ➜ **`/full-text`**: Returns the full plain text of a document specified by a document ID

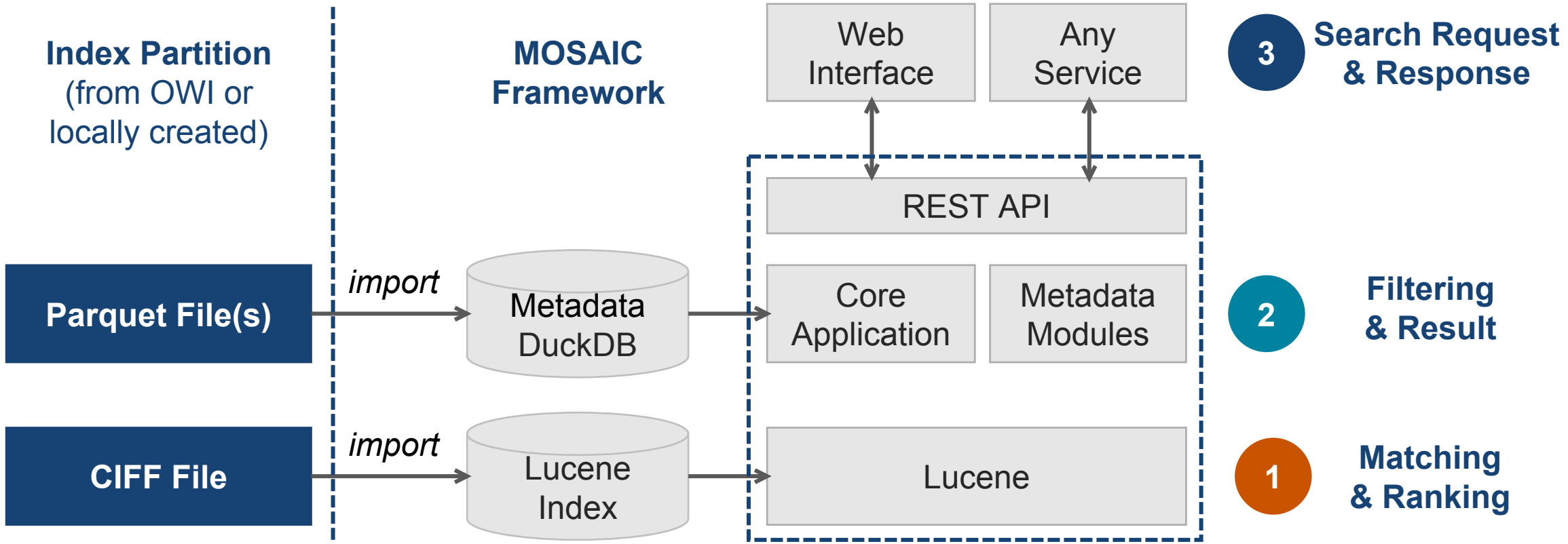
Retrieval Process of MOSAIC (simplified)



Development Possibilities

(1) Create your own index

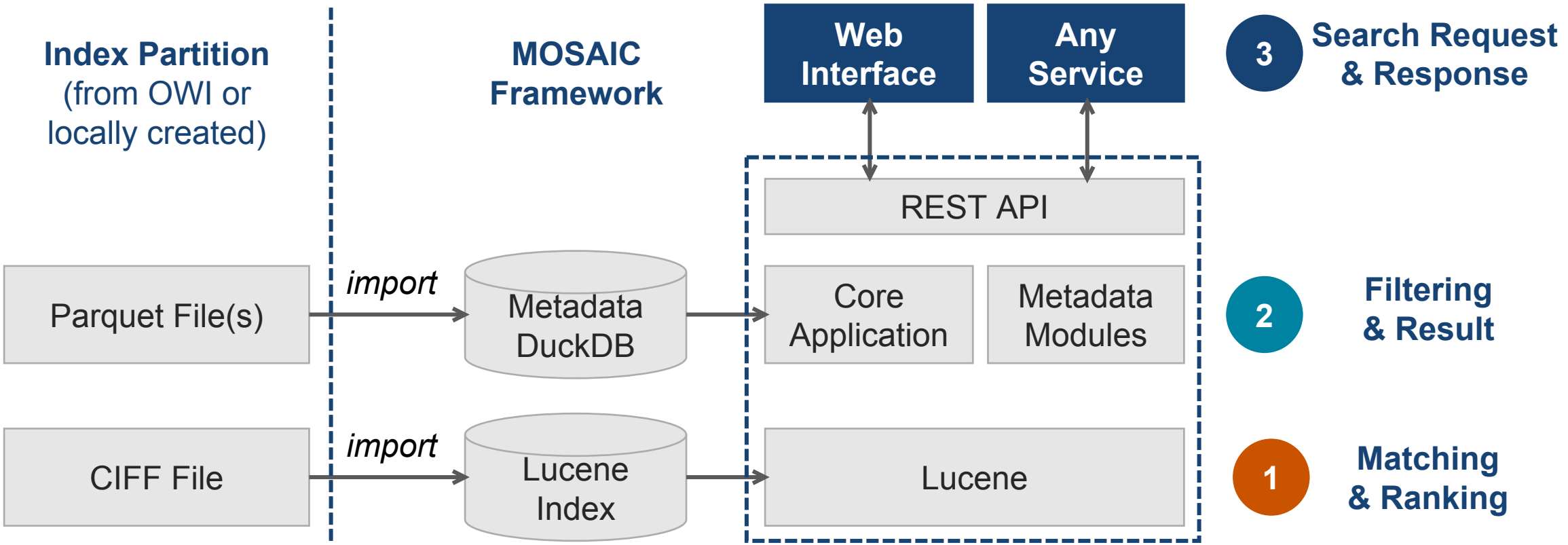
- Move/Copy CIFF and Parquet file(s) to directory `resources/<INDEX_NAME>/`
- Serve CIFF/Lucene and Parquet file(s) using CLI options



Development Possibilities

(2) Create or improve a front-end

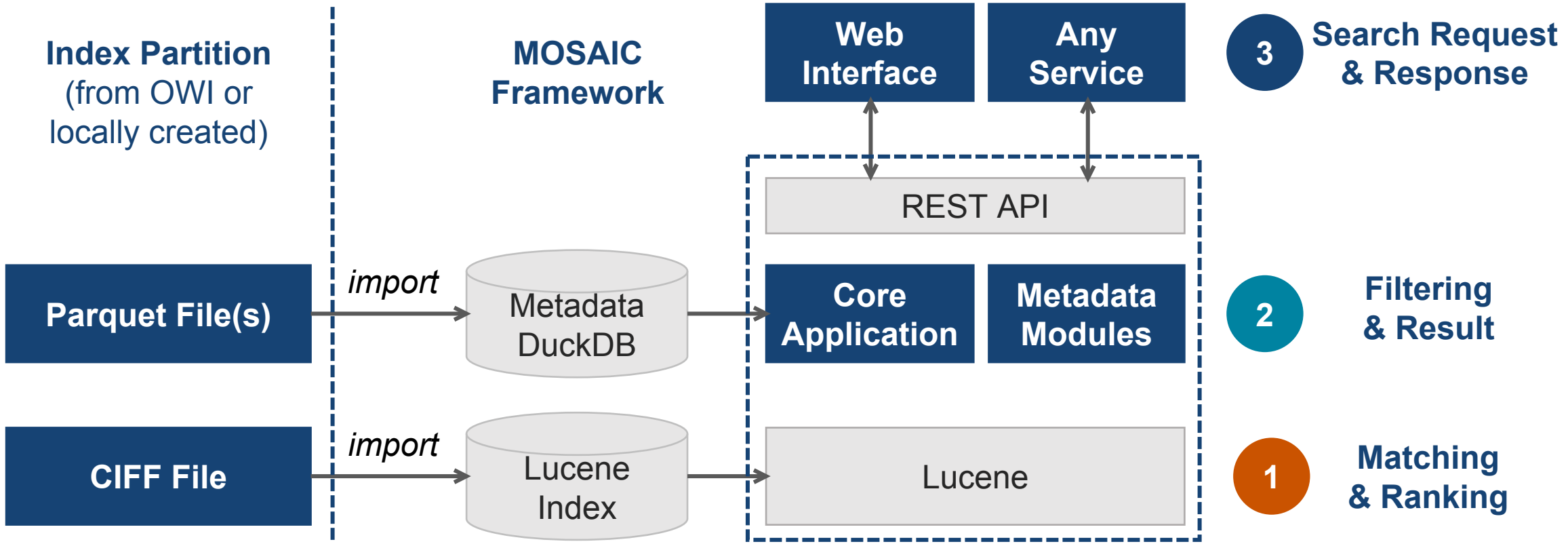
- Modify the existing front-end in the directory `front-end` of the repository
- Create a new (web) user interface utilizing the REST API



Development Possibilities

(3) Create an application that uses MOSAIC as service

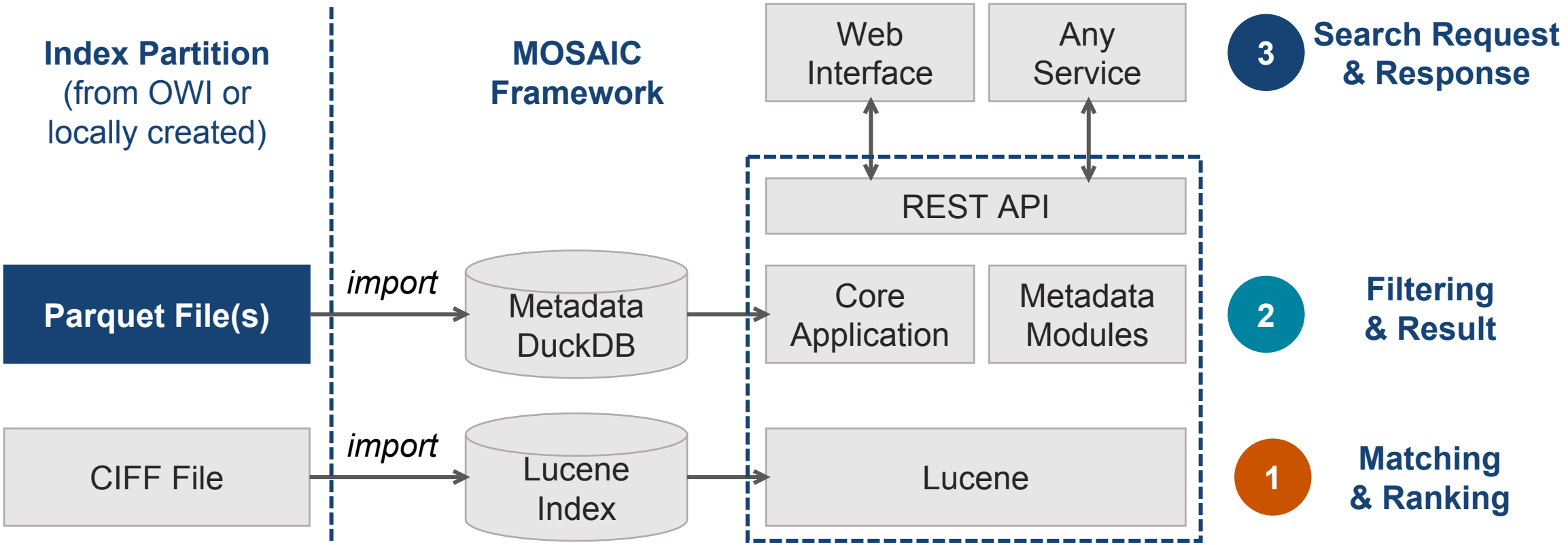
- Integrate a new index into MOSAIC (1)
- Modify or create a front-end (2)
- Access the REST API and use MOSAIC as service



Development Possibilities

(4) Undertake web data analysis

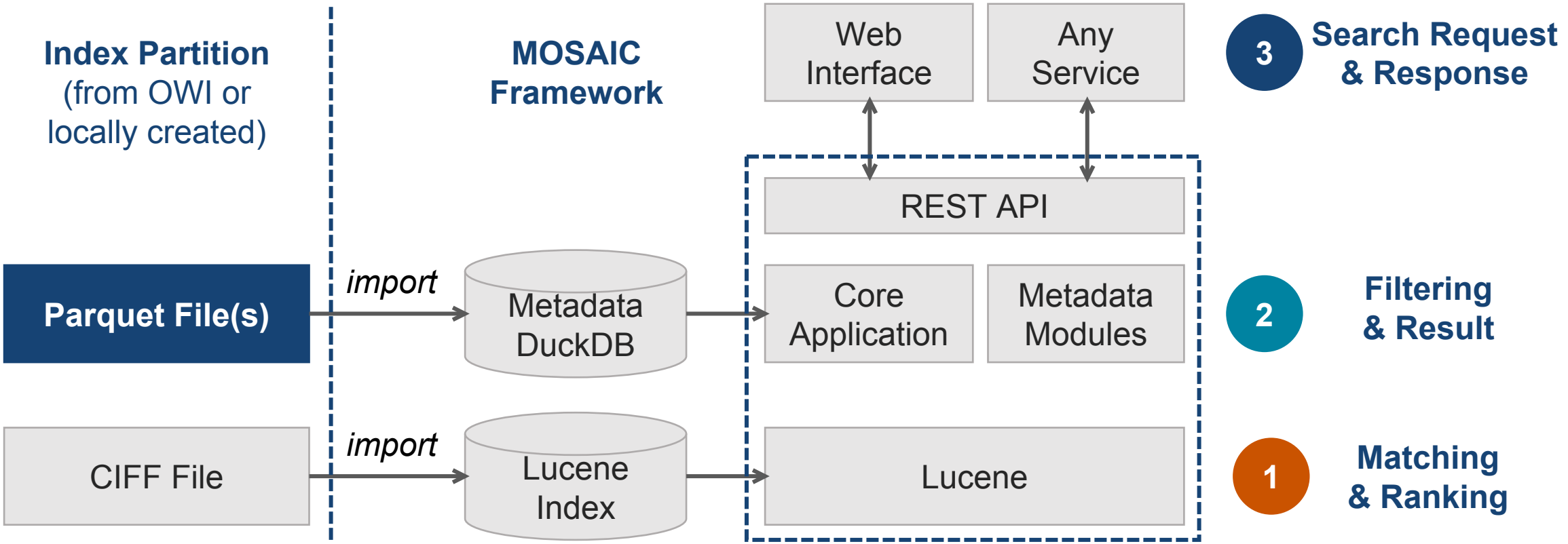
- Use existing and/or locally created indices
- Perform analysis on set of web documents
- Example: Topic Modeling, PageRank implementation, etc.



Development Possibilities

(5) Feed a Large Language Model

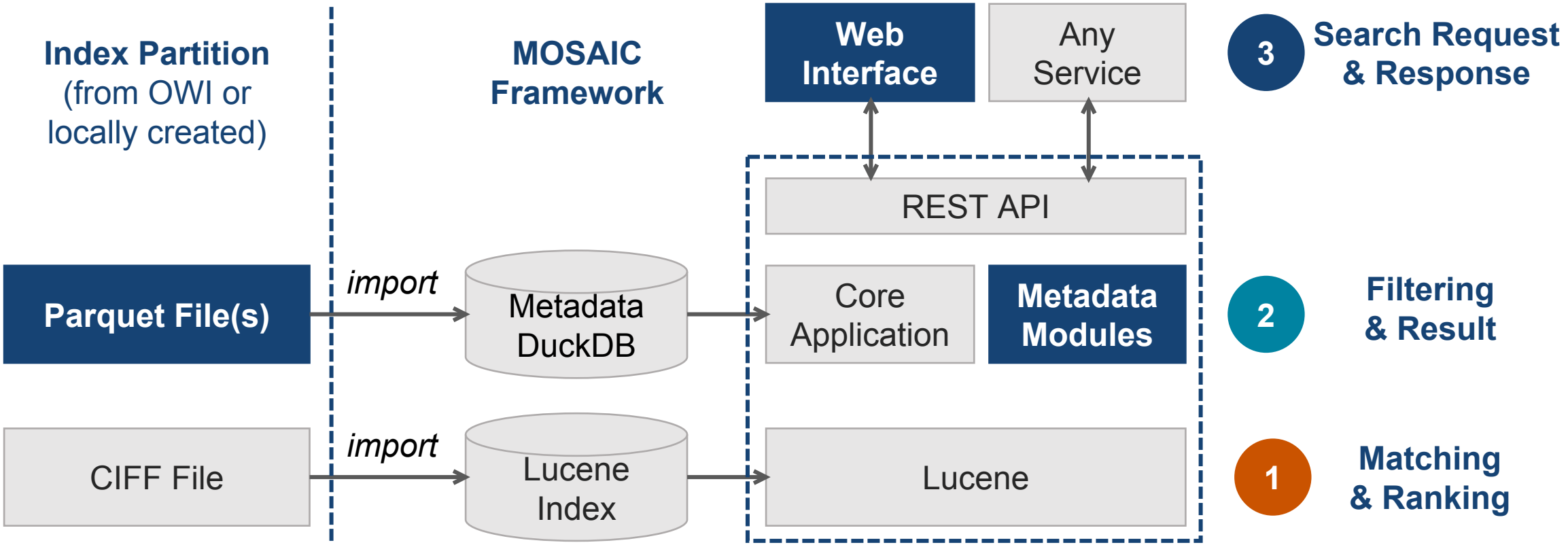
- Use existing and/or locally created indices
- Develop use case and analysis goal



Development Possibilities

(6) Create a new module

- Follow steps in Developer Guide for creating a new module
- Example: Make use of existing unused column `domain_label` in Parquet file



Concrete Ideas & Suggestions

- ➜ Create a new front-end and utilize the REST API of MOSAIC
- ➜ Create a new module using existing but unused metadata columns (e.g., `domain_label`) in the Parquet file
- ➜ Expand the existing `/index-info` endpoint byperforming an in-depth analysis of web documents


TU Graz
WISSEN TECHNIK LEIDENSCHAFT
www.tugraz.at
SCIENCE PASSION TECHNOLOGY
GRAZ UNIVERSITY OF TECHNOLOGY AUSTRIA

Any Questions?

Contact:

- Sebastian Gürtl <sebastian.guertl@tugraz.at>
- Alexander Nussbaumer <alexander.nussbaumer@tugraz.at>
- Graz University of Technology, Austria

Agenda

- ➜ 09:45 - 10:00: Group formation and coffee
- ➜ 10:00 - 11:00: Working session 1: Idea generation
- ➜ 11:00 - 11:15: Presentation of ideas (*)
- ➜ 11:15 - 13:00: Working session 2
- ➜ 13:00 - 14:00: Lunch
- ➜ 14:00 - 17:00: Working session 3
- ➜ 17:00 - 18:00: Presentation of results & voting (*)
- ➜ 18:00 - 19:00: Get-together and beer

* Plenary session, will be broadcasted to online participants