

# LLM-READY SPATIO-TEMPORAL DATA

## ENABLING AGENTIC GEOAI WITH STAC

Pieter Kempeneers

Chiara Chiarelli<sup>1</sup>, Pieter Kempeneers<sup>1</sup>, Ionut Trandafir<sup>2</sup>

1 European Commission, Joint Research Centre (JRC), 2 Unisystems

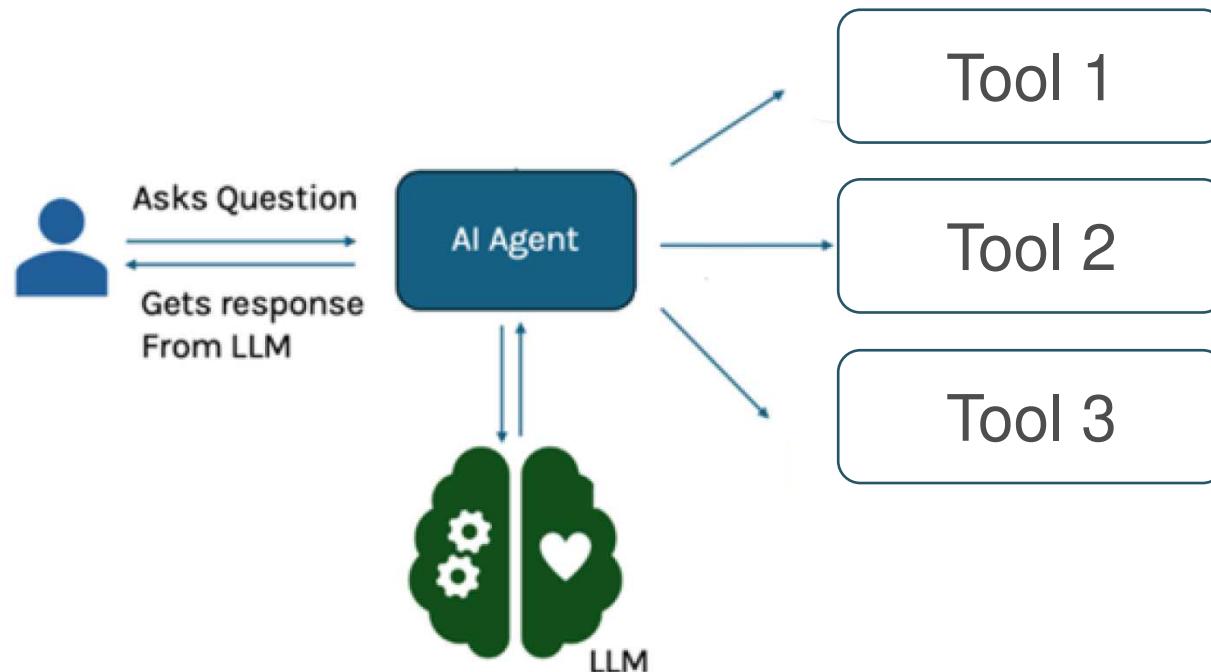
October 2, BiDS'25, Riga



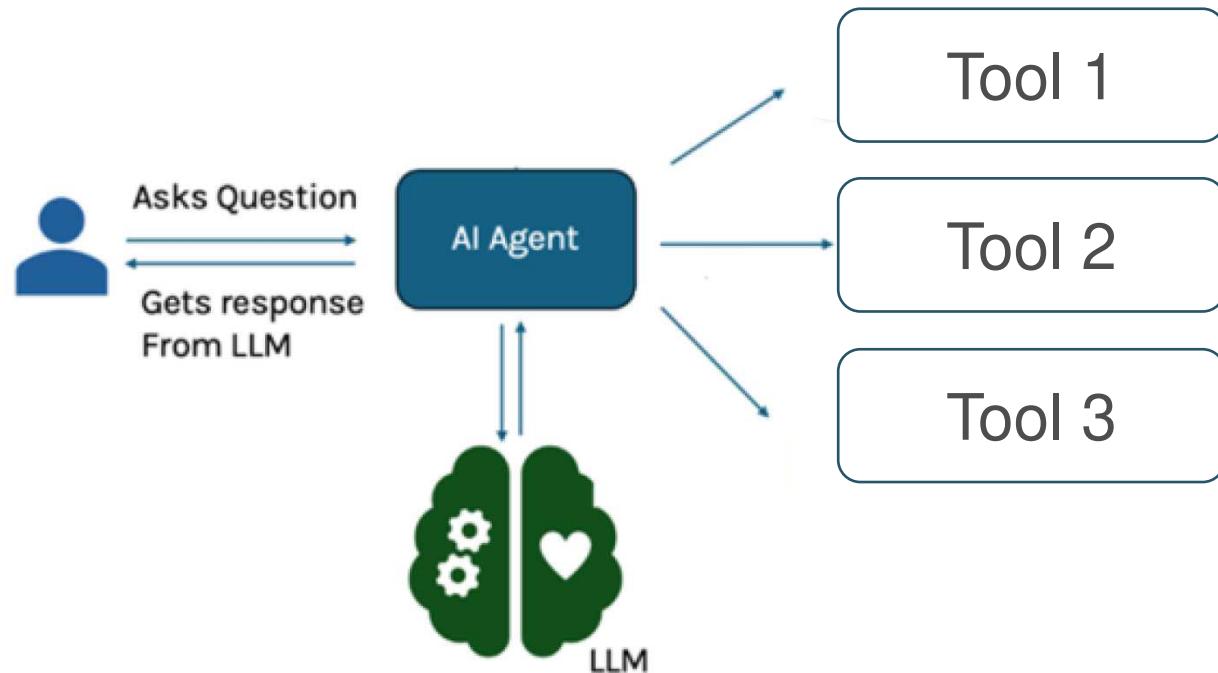
# Content

- Objective
- GeoAI architecture
- Experiment
- Lessons learned
- Conclusions

# Objective: enabling agentic GeoAI with STAC

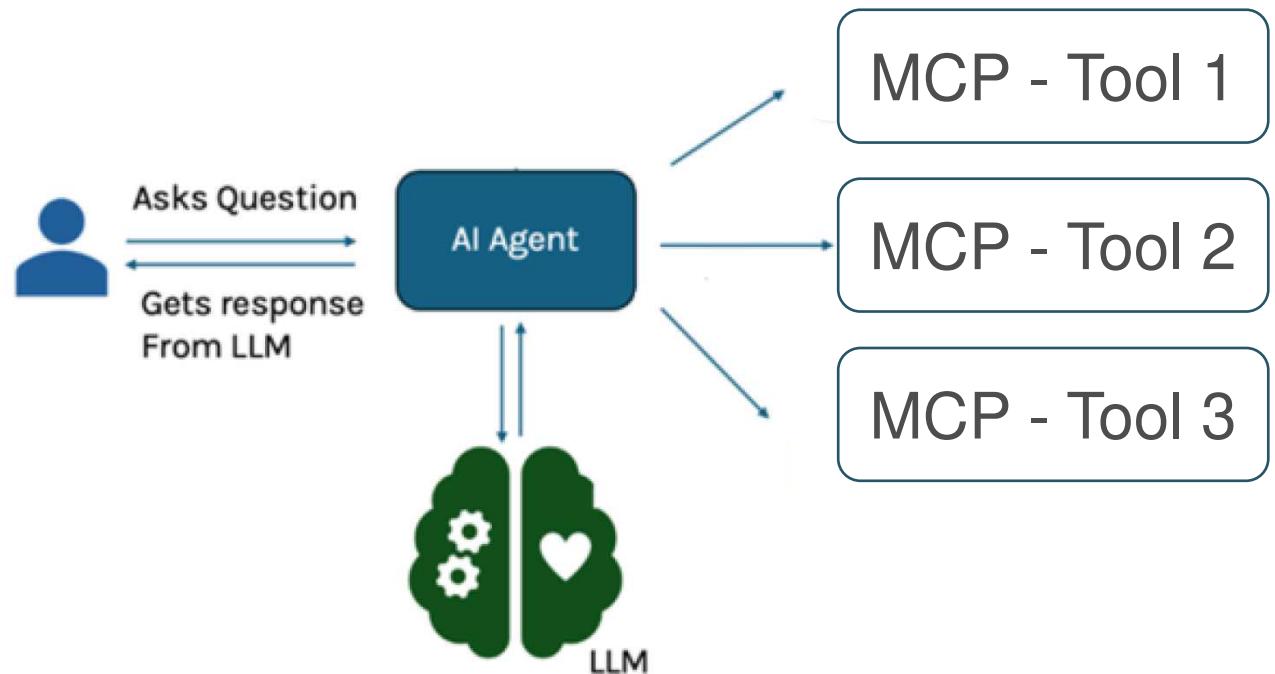


# GeoAI based on agentic AI



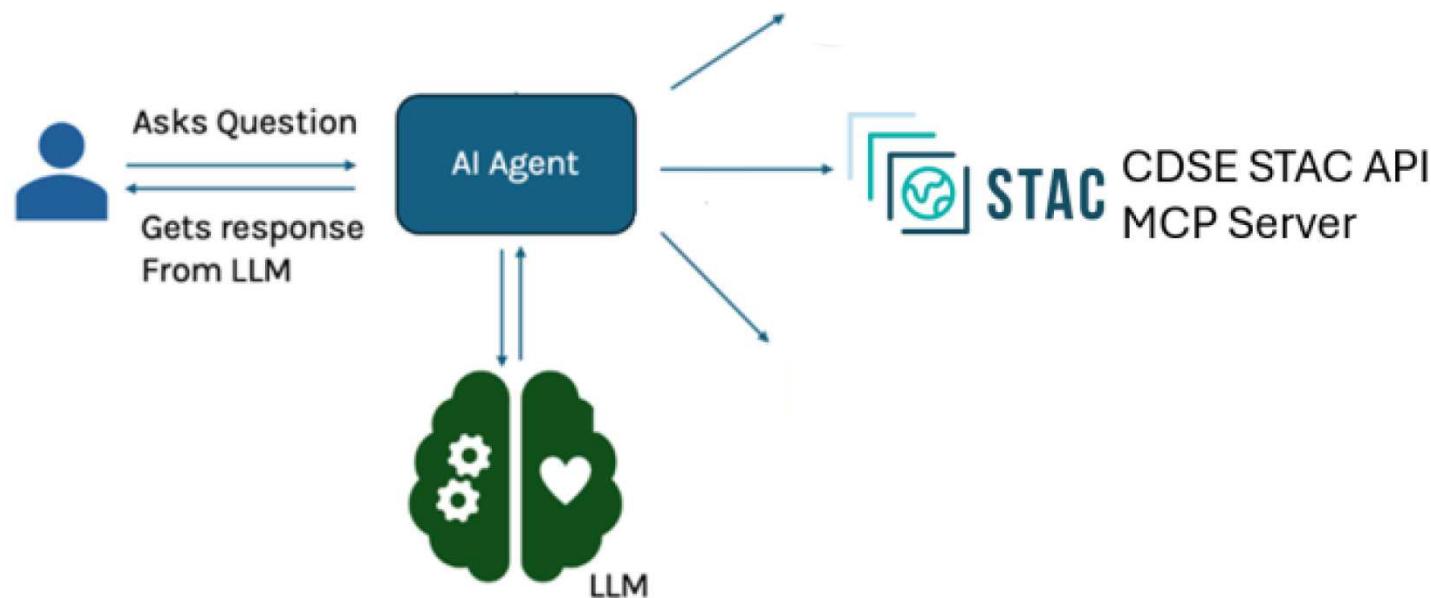
**Tool chaining**  
Orchestrate multiple tools in sequence, breaking complex task into subtasks

# Tool access via Model Context Protocol (MCP)



**Model Context Protocol**  
Enable AI models and applications to interact with external data sources and tools in a consistent manner

# STAC MCP server



**STAC MCP server**  
Use natural language to search and access all STAC data through intuitive queries

# SpatioTemporal Asset Catalog (STAC)

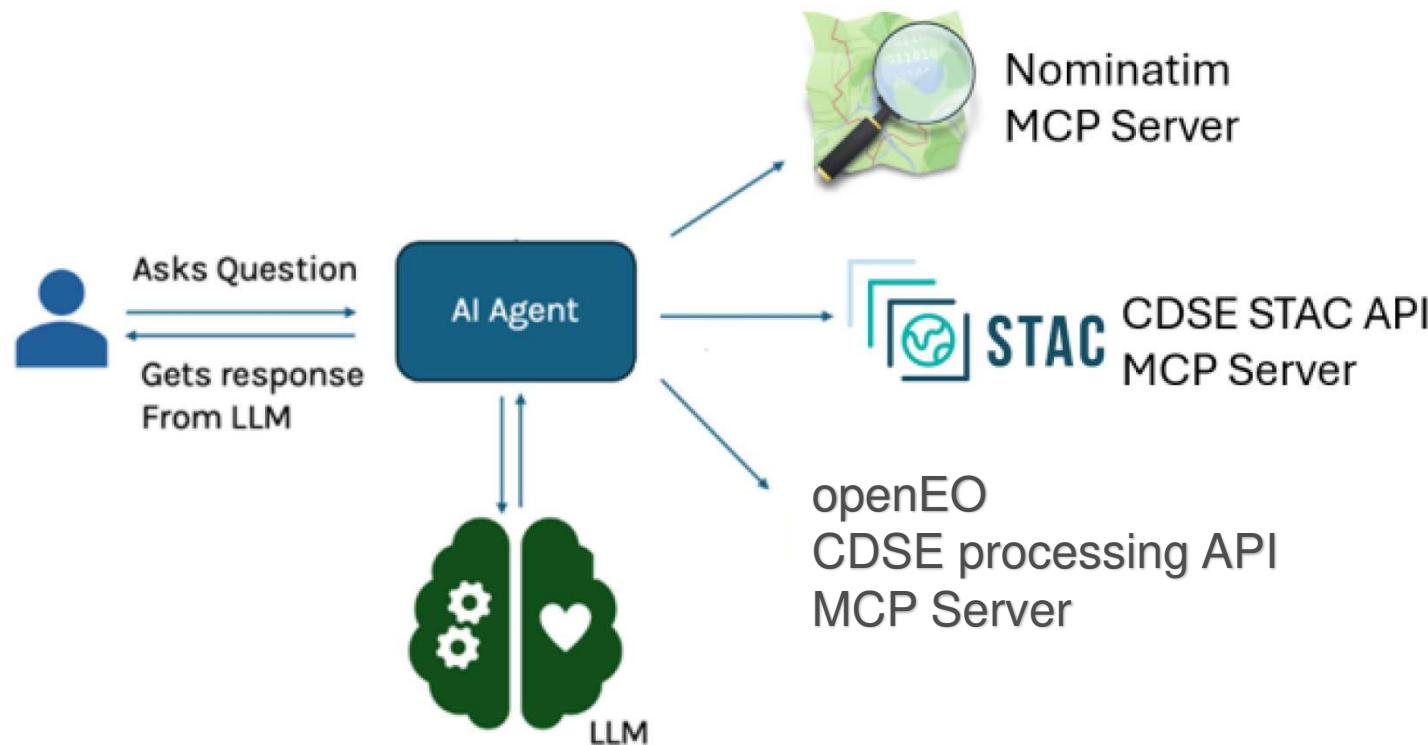
- Open specification for **describing geospatial information**;
- Provides a **common structure** to organize and search spatiotemporal assets;
- Facilitates working with Earth Observation **data from different providers**.

# STAC MCP server: key features

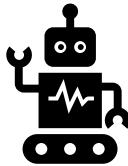
- **Universal STAC Compatibility:** works with any STAC-compliant API (Copernicus Data Space Ecosystem, AWS Earth Search, Microsoft Planetary Computer, etc.)
- **MCP Protocol Support:** fully compatible with MCP clients and AI applications
- **Comprehensive Search:** advanced search capabilities with spatial, temporal, and metadata filtering
- **Collection Management:** discover available satellite datasets
- **Docker Ready:** easy deployment with Docker and Docker Compose
- **Configurable:** flexible configuration through environment variables

Released as open source: <https://code.europa.eu/jrc-bdap/stac-api-mcp-server>

# GeoAI architecture accessing MCP servers



# Experiment: Geo-AI agent



What can you do?



I can help you with a variety of data analytics and processing tasks. I have access to tools that allow me to:

**Geocode addresses:** ...

**Search for satellite imagery:** ...

**Get information about STAC collections and items:** ...

**Perform OpenEO processes:** ...

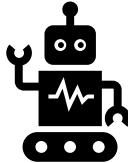
**Fetch web page content:** ....

**Get the current date:** ...

**Perform Google searches:** ...

What kind of data task are you looking to accomplish today?

# Geo-AI agent



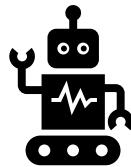
What is the most cloud free image over Ispra in Italy acquired in June 2025, based on Sentinel-2?

The most cloud-free Sentinel-2 image over Ispra, Italy, acquired in June 2025, is:

- Item ID:  
S2B\_MSIL2A\_20250622T102559\_N0511\_R108\_T32TMR\_20250622T  
124912
- Acquisition Date: 2025-06-22 at 10:25:59 UTC
- Cloud Cover: Approximately 0.46%

Would you like to get more details about this image or perform any other task?

# Geo-AI agent



Please calculate a vegetation index image over this area

I have successfully authenticated with the OpenEO backend. Now, I will create an NDVI job for the specified area and date.

...

I have successfully created an NDVI calculation job with the ID cdse-j-2509181658554c02913e338a106153d1.

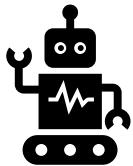
...

I will start the NDVI job.

...

The NDVI job cdse-j-2509181658554c02913e338a106153d1 has been successfully started and is currently queued.

# Geo-AI agent



What is the job status?

Here is the status of your job:

NDVI Job (cdse-j-2509181658554c02913e338a106153d1):

Status: finished

Progress: 100%

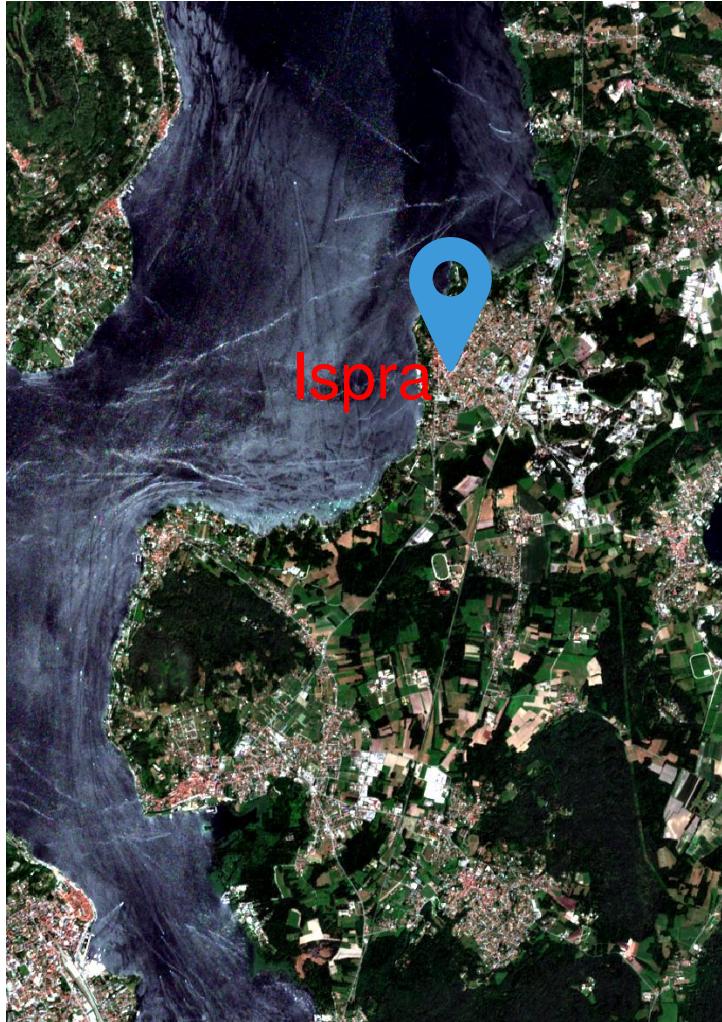
Title: NDVI for Ispra, Italy - June 22, 2025

Description: NDVI calculation for {'west': 8.55, 'south': 45.75, 'north': 45.85, 'east': 8.65} from 2025-06-22 to 2025-06-22

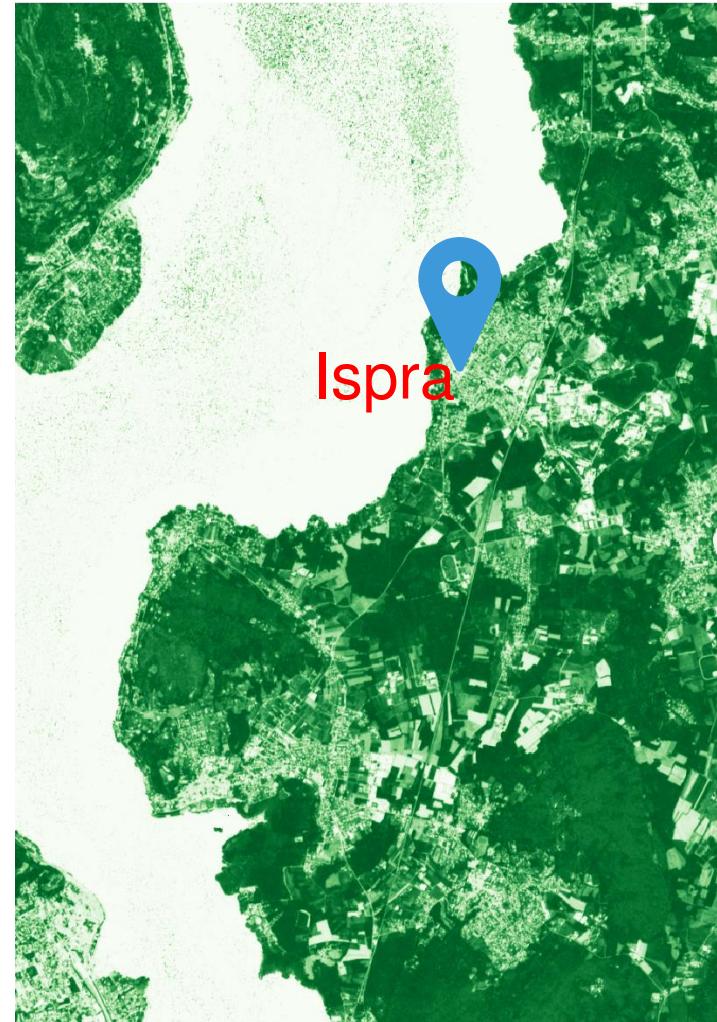
The NDVI job is complete! You can now get its results.

# Geo-AI agent

True color image (RGB)



Vegetation index (NDVI)



# Lessons learned

## Issue

- **Excessive token usage** frequently reaching the LLM's token limit;
- Geospatial queries lead to **imprecise spatial representations** and introduced unnecessary steps into the workflow;
- **Inefficient and redundant API calls**, emphasizing the need for optimized endpoints.

## Solution

- **STAC API extensions, Sort and Fields, reduce token usage** by limiting metadata requests to essential fields and ordering results by relevance;
- **Integrate polygon geocoding** through the Nominatim tool to simplify preprocessing and reduce query complexity by directly retrieving accurate GeoJSON polygons.

# Conclusions

- Prototype demonstrates how an LLM agent can **retrieve, and process geospatial data exposed by a STAC catalog, using MCP-enabled tools**;
- Feasibility study uses relatively simple query. Further investigation is required to evaluate performance with **more complex and diverse queries**;
- Future work: explore adaptive interaction strategies, **multi-agent orchestration**, and enhanced tooling for image analysis, **visualization** and batch processing.