Graph-Grounded Agentic Search for Biodiversity e-Infrastructures: a vision for combining LLMs and Semantic Web Technologies in LifeWatch ERIC

Lucia Vaira[1,*,†], Nikos Minadakis[1,†], Ilaria Rosati[2,†], Eftychia Tzafesta[1,†], Martina Pulieri[3,†], Houda Ben Salah[1,†] and Alberto Basset[1,4,†]

[1]LifeWatch ERIC Service Centre, Campus Ecotekne, Lecce, Italy
[2]National Research Council (CNR), Research Institute on Terrestrial Ecosystems (IRET), Lecce, Italy
[3]Department of Earth and Marine Science (DiSTeM), University of Palermo, Palermo, Italy
[4]Department of Biological and Environmental Sciences and Technologies, University of Salento, Lecce, Italy

Abstract

Biodiversity and ecosystem research increasingly depends on the ability to discover, combine, and interpret heterogeneous digital resources. There already exist research infrastructures providing FAIR catalogues for biodiversity and ecosystem domain, yet users still face practical barriers when translating complex scientific information needs into structured searches over metadata, vocabularies, spatial and temporal constraints, provenance, licenses, and analytical capabilities. This paper presents a graph-grounded agentic framework for complex structured search in LifeWatch ERIC. The approach uses Large Language Models (LLMs) not as unconstrained answer generators, but as query-planning agents that decompose natural-language requests into semantic slots, ground them against ontologies and controlled vocabularies, generate structured queries over catalogue and knowledge-graph layers, and validate intermediate outputs using schema and consistency constraints. We describe a conceptual architecture that combines LLM-based planning, Semantic Web standards, knowledge graph retrieval, SHACL-style validation, and provenance-aware explanations, which acts as a reusable design pattern for trustworthy LLM-assisted discovery in biodiversity e-infrastructures, together with competency questions, failure modes, and an evaluation protocol covering semantic grounding accuracy, query-plan correctness, retrieval relevance, and explanation faithfulness.

Keywords

Large Language Models, Agentic AI, Semantic Web, Knowledge graphs, FAIRness, Scientific search, LifeWatch ERIC, LifeWatch Italy, Graph-RAG, Research Infrastructures

1. Introduction

Scientific discovery in biodiversity and ecosystem research increasingly resembles navigation through a living archipelago of digital objects. Researchers need to locate assets like datasets, services, workflows, instruments, and semantic artefacts, and that are distributed across institutions, disciplines, and governance boundaries. The FAIR principles emphasise that data and metadata should be findable, accessible, interoperable, and reusable for both humans and machines [1]. In practice, however, the act of finding the right resource is rarely a simple keyword lookup.

To make a very concrete example, a researcher may ask for marine biodiversity datasets for a specific region and period, require taxonomic normalisation, prefer resources with open licenses, and need an associated workflow that supports reproducible analysis. Such a request is semantically rich, multi-constraint, and often only partially specified. LifeWatch ERIC (https://lifewatch.eu) offers an important

Graph-enhanced LLMs for trustwOrthy Web data management (GLOW), in ISWC 2026 Workshops Joint Proceedings, October 25–26, 2026, Bari, Italy

*Corresponding author.
†These authors contributed equally.
lucia.vaira@lifewatch.eu (L. Vaira); nikos.minadakis@lifewatch.eu (N. Minadakis); ilaria.rosati@cnr.it (I. Rosati); eftychia.tzafesta@lifewatch.eu (E. Tzafesta); martina.pulieri@unipa.it (M. Pulieri); houda.belsalah@lifewatch.eu (H. Ben Salah); alberto.basset@lifewatch.eu (A. Basset)
0000-0002-5935-6074 (L. Vaira); 0000-0003-3422-7230 (I. Rosati); 0000-0003-3330-2163 (E. Tzafesta); 0009-0001-9691-1989 (M. Pulieri); 0000-0002-3603-9316 (A. Basset)

© 2026 Copyright © 2026 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-11-05

CEUR-WS.org/Vol-4313/glow-paper2.pdf

setting for this problem. It is a European e-science research infrastructure consortium for biodiversity and ecosystem research of eight European Union Member States and three Common Facilities, the core components of the infrastructure. Member countries operate from local Distributed Centres (as national nodes) assisting LifeWatch ERIC in several concrete ways via in-kind contributions such as resource hosting, data provision and integration, software development, and community engagement, acting as a liaison to the local ecological research community, handling user support, training, and outreach. Among the Distributed Centres, the Italian one (i.e., LifeWatch Italy), coordinated by the National Research Council of Italy, actively collaborates with LifeWatch ERIC on all thematics related to (meta)data management and semantics. LifeWatch ERIC indeed provides access to scientific services, reproducible analytics, and community-oriented digital environments via its Metadata Catalogue (available at https://metadatacatalogue.lifewatch.eu), designed to manage and publish metadata about biodiversity and ecosystem domain related to six kinds of resources, i.e., datasets, services, Virtual Research Environments (VREs), research sites, workflows, and training resources through standard-based metadata management [2]. These assets create fertile ground for machine-actionable discovery, but they also expose a familiar infrastructure paradox: the richer the metadata ecosystem becomes, the harder it can be for non-specialist users to express their needs in the precise language of catalogues, schemas, ontologies, and APIs. And here is where Large Language Models (LLMs) can help, by acting as a powerful natural-language interface aiming at translating and reformulating human queries into actionable system commands and structured data. LLMs can infer missing context and produce fluent explanations. Yet LLMs also remain vulnerable to weak provenance and insufficient explainability, particularly in scientific settings where a plausible answer can be more dangerous than no answer [3]. Initially focused on domain-specific tasks, LLMs have evolved through increasingly sophisticated architectures and training methods, enabling broader, more refined language understanding and interaction that extend beyond traditional NLP applications [4, 5]. The core argument of this paper is therefore intentionally narrow: LLMs should not replace structured metadata and semantic infrastructure. Instead, they should orchestrate discovery over it. In this pattern, an LLM acts as an agentic query planner whose outputs are grounded, checked, and constrained by Semantic Web representations. The paper is organised as follows: section 2 presents the background, the problem setting and the motivating scenario along with the competency questions. Section 3 presents the graph-grounded agentic search framework, which allows to transform natural-language information needs into auditable search plans, align user terms with controlled vocabularies and ontologies, generate structured queries over catalogue and knowledge-graph layers, validate results against semantic constraints, and return provenance-aware explanations. Section 4 briefly introduces the evaluation methodology and finally, Section 5 presents the future developments and concludes the paper.

2. Background and problem setting

LifeWatch ERIC operates in a domain where data heterogeneity is structural rather than accidental. Biodiversity resources may describe species occurrences, environmental observations, genomic measurements, ecosystem functions, marine or terrestrial sites, analytical workflows, and software services. LifeWatch ERIC endorses, adopts, and promotes the data systems within its Thematic Services (https://www.lifewatch.eu/thematic-services/) that integrate all available biodiversity digital resources, which mainly serve as transnational frameworks that provide data and services to test specific hypotheses and address scientific questions. Semantic Web technologies and standards are a natural backbone for this kind of landscape. In order to provide a common setting to understand the main terminology, here below the list of technologies and standards that will be mentioned throughout the paper: RDF provides a graph-based data model for representing Web information [6]; SPARQL supports expressive querying over RDF graphs [7]; DCAT provides a vocabulary for interoperable catalogue descriptions of datasets and data services [8]; SKOS supports the publication and linking of controlled vocabularies, thesauri, taxonomies, and classification schemes [9]; PROV-O models provenance, enabling systems to represent how entities, activities, and agents relate to one another [10]; GeoSPARQL supports representation and

querying of geospatial data on the Semantic Web [11]; SOSA/SSN can describe sensors, observations, samples, and related measurement contexts [12]; and SHACL provides a language for describing and validating RDF graph constraints [13]. The challenge is that these standards are powerful but not always easy for users to invoke. A biologist may express a need through ecological concepts, project names, taxonomic labels, regions, time periods, data quality expectations, and intended analysis. A catalogue, by contrast, expects structured fields, normalised identifiers, spatial geometries, temporal intervals, controlled resource types, and often service-specific parameters. This mismatch creates an interaction bottleneck that is especially visible when search becomes compositional. Users are not merely looking for one dataset, they are assembling a research-ready bundle of resources that can support a reproducible scientific task. Retrieval-augmented generation (RAG) addresses part of this problem by connecting language models to external stores [14]. Recent work on LLMs and knowledge graphs argues that the two technologies are complementary: LLMs provide language understanding and generalisation, while knowledge graphs provide explicit, structured, and interpretable knowledge [14]. For research infrastructures, however, the key question is not only how to retrieve text passages for generation. It is how to plan, execute, and explain structured discovery actions across metadata catalogues and semantic artefacts catalogues. This motivates a graph-grounded agentic approach, which instead of relying on simple keyword matching, treats user requests as structured competency questions (CQs) to better capture spatial, temporal, and taxonomic nuances. Consider the following informative need: "*Find marine biodiversity resources for analysing changes in plankton community composition in the Adriatic Sea after 2015, preferably with taxonomic name alignment and a reproducible workflow*". A standard keyword search may retrieve records containing "plankton", "Adriatic", or "workflow", but it is unlikely to expose the hidden structure of the request. The user is asking for resource type matching, taxonomic semantics, spatial filtering, temporal filtering, service compatibility, analytical intent, and reproducibility constraints. Representative CQs for the search layer include: 1) Which datasets concern a target taxonomic or ecological concept, after expansion through relevant thesauri, taxonomies, or ontology mappings? 2) Which resources overlap a user-defined spatial region and temporal interval? 3) Which services or workflows can operate on the discovered datasets, given input type, format, license, access, and methodological constraints? 4) Which retrieved resources provide sufficient provenance, citation, license, and quality metadata for reuse? 5) Which parts of the answer are supported by the metadata catalogue, which by semantic mappings, and which by model inference? 6) Which candidate results fail validation, and why? The broad query is hence decomposed into logical steps (that correspond to the above listed CQs), ensuring that the retrieved data is both reproducible and verifiable. These questions also prevent the LLM from jumping directly from a broad user request to a polished but unverifiable answer. The agent must instead translate, reason, and expose a plan. Then it should plan elements to semantic resources, execute constrained retrieval, and report the evidence trail for its findings.

3. Graph-grounded agentic search framework

The proposed framework consists of six layers: user interaction, agentic planning, semantic grounding, structured retrieval, validation and ranking, and explanation. Figure 1 summarises the pipeline by specifying the function and output of each layer.

First, the **user interaction layer** receives natural-language scientific search requests. The system encourages clarification but does not require the user to know catalogue schemas or ontology identifiers. Second, the **agentic planning layer** decomposes the request into a structured search plan. This plan contains semantic slots such as target organism or ecological concept, observed property, ecosystem type, spatial footprint, temporal interval, resource type, required service capability, license, provenance requirement, and intended analysis. The LLM may propose candidate interpretations, but each interpretation is marked as tentative until grounded.

Third, the **semantic grounding layer** links the extracted slots to machine-actionable identifiers. Domain terms are mapped to SKOS concepts, ontology classes, taxonomic identifiers, DCAT resource

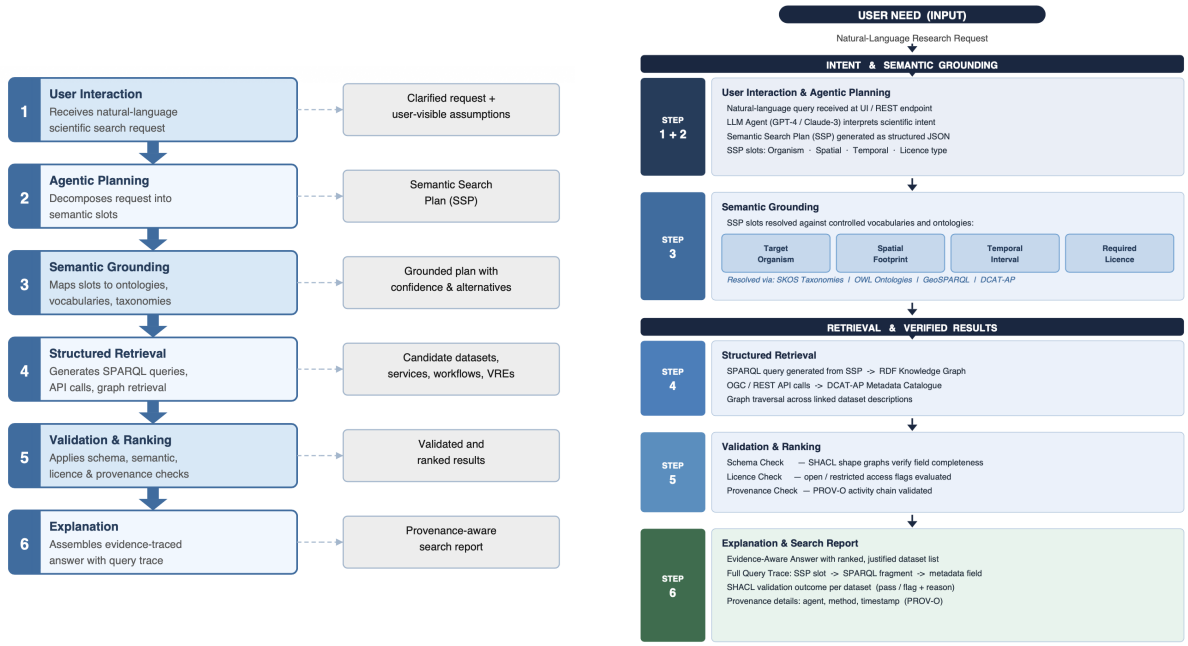


Figure 1: Main layers (on the left) and conceptual architecture (on the right) of the proposed graph-grounded agentic search framework.

types, GeoSPARQL geometries, temporal intervals, and service capability descriptors. Ambiguous mappings are retained as alternatives rather than silently collapsed. Fourth, the **structured retrieval layer** translates the grounded plan into catalogue queries, SPARQL queries, API calls, or hybrid graph-vector retrieval actions. Fifth, the **validation and ranking layer** checks whether retrieved candidates satisfy the semantic contract of the plan. SHACL shapes can validate required metadata fields, resource type consistency, license presence, spatial or temporal completeness, service input/output compatibility, and provenance availability. Finally, the **explanation layer** returns results with the plan, evidence, validation outcomes, and unresolved assumptions. To ground this more concretely, consider how the six layers handle the informative need expressed in Section 2. The request "*find marine biodiversity resources on plankton community changes in the Adriatic Sea after 2015*" enters Layer 1 as plain text. Layer 2 breaks it down into semantic slots: ecological concept, spatial extent, temporal bound, resource types, and reusability constraints. Layer 3 maps "plankton community composition" to a controlled vocabulary term, resolves "Adriatic Sea" to a GeoSPARQL geometry, and converts "after 2015" into a formal temporal lower bound. Layers 4 and 5 execute structured retrieval and validate candidates against the plan, while Layer 6 returns a provenance-aware bundle rather than a raw list of hits. This architecture changes the role of the LLM. The model is not the authority on which resource exists or whether it is reusable. The graph and catalogue are the authority. The LLM is a planner, translator, and dialogue manager operating under semantic supervision. In the small, this is a user-interface improvement. In the large, it is an infrastructure pattern for trustworthy AI-mediated scientific discovery.

3.1. Semantic Search Plan representation

A central artefact of the framework is the Semantic Search Plan (SSP), a machine-readable intermediate representation produced by the agent and validated before execution. The SSP acts as a contract between natural language and infrastructure search. It is deliberately explicit, because hidden prompts are poor substitutes for inspectable scientific workflows. An SSP may include fields such as intent, domain concept, observed variable, organism or taxon, ecosystem, spatial constraint, temporal constraint, resource class, data format, access condition, license, provenance requirement, service or workflow requirement, and ranking preference. Each field contains a value, grounding candidates, confidence, provenance of grounding, and validation status.

Considering the previously defined informative need: "*Find marine biodiversity resources for analysing changes in plankton community composition in the Adriatic Sea after 2015, preferably with taxonomic name alignment and a reproducible workflow*", the "plankton community composition" may be represented as an ecological concept with mappings to controlled vocabulary terms, while "Adriatic Sea" may be represented as a GeoSPARQL-compatible region or as a named feature resolved through a gazetteer. "After 2015" becomes a temporal lower bound. "Reproducible workflow" becomes a requirement over workflow metadata, executable service descriptions, or VRE availability. The SSP also encodes uncertainty. If a term maps to several possible concepts, the system can either ask the user to choose or run parallel constrained searches and display the effect of each interpretation. This is important in biodiversity, where taxonomic synonyms, common names, ecological groupings, and measurement variables may vary across communities. The point is not to eliminate ambiguity at the language layer but to make it visible before it contaminates retrieval.

A simplified SSP for the motivating scenario is shown below:

```
intent: discover_research_bundle
domain: marine biodiversity
concept: plankton community composition
location: Adriatic Sea [grounded as named marine region or geometry]
time: start >= 2015-01-01
resource_types: dataset, service, workflow, VRE
constraints: taxonomic alignment available; provenance present; reusable license
    preferred
ranking: semantic match, metadata completeness, workflow compatibility, recency
explanation_required: true
```

Being the SSP structured, it can be stored, compared, evaluated, and reused. It can also be exposed to expert users who want to edit the plan before execution. In this sense, the SSP is a lightweight research object for search itself.

3.2. Query generation and retrieval over catalogue and graph layers

Once an SSP has been validated, the framework generates retrieval actions. These may include keyword queries against catalogue indexes, structured filters over metadata fields, SPARQL queries over an RDF representation, and graph-guided retrieval for related resources. The retrieval layer should support hybrid search because catalogue ecosystems rarely expose all relevant information in one formalism. For example, a dataset metadata record may be indexed in the LifeWatch ERIC Metadata Catalogue, its semantic annotations may be represented in RDF, its service compatibility may be documented in an API registry, and its workflow relation may be captured in a VRE or workflow management system. In an RDF-oriented deployment, DCAT can describe datasets, data services, catalogues, distributions, contact points, themes, licenses, and access URLs. SKOS can represent controlled themes and broader/narrower concept expansion. PROV-O can express resource lineage and responsible agents. GeoSPARQL can represent spatial footprints and spatial relations. SOSA/SSN can represent observations and measured properties when observation-level semantics are available. SHACL shapes then provide an executable gatekeeper, checking whether candidate resources satisfy expected structural conditions. An illustrative SPARQL-like query generated from the SSP might retrieve resources typed as datasets, annotated with a plankton-related concept or one of its narrower concepts, spatially intersecting the Adriatic Sea geometry, and temporally beginning after 2015. A second query may retrieve workflows or services whose declared input requirements match the dataset distributions. A third query may retrieve provenance and license metadata. The system then composes a ranked bundle rather than a flat list of disconnected records. Concretely, for the Adriatic Sea scenario of the informative need, the first retrieval action might look like this:

```
SELECT DISTINCT ?dataset ?title ?period WHERE {
  ?dataset a dcat:Dataset ;
           dct:title ?title ;
```

```
          dcat:theme ?concept ;
          dct:spatial ?region ;
          dct:temporal ?period .
  ?concept skos:broader* :PlanktonCommunityComposition .
  FILTER ( geof:sfIntersects(?region, :AdriaticSeaGeometry) )
  FILTER ( ?period >= "2015-01-01"^^xsd:date )
}
```

A follow-up query retrieves workflows whose declared input type matches the dataset distributions, and a third collects provenance and license metadata. Every result of the final bundle carries a trace back to the SSP slot and the query fragment that produced it. The retrieval process should indeed produce a trace. Each candidate result is associated with the query fragment that retrieved it, the semantic grounding used, validation results, and ranking rationale. This trace is the basis for user-facing explanation and for later evaluation. It also enables expert users to diagnose failures: perhaps a relevant dataset was excluded because its temporal coverage was missing, or a workflow was not correctly evaluated because no machine-readable input format was declared.

3.3. Validation, trust, and hallucination control

Trustworthy search requires more than better prompts. In the proposed framework, hallucination control is handled through architectural separation: the LLM may propose a plan, but the plan is checked against known schemas and vocabularies; the LLM may draft a query, but the query is executed against authoritative catalogue and graph layers; the LLM may summarise results, but each statement in the summary is linked to retrieved metadata and validation evidence. The validation layer applies several guards: i) Schema validation checks that required fields are present and correctly typed. ii) Semantic validation checks that resource classes, themes, variables, and service capabilities are compatible with the plan. iii) Spatial-temporal validation checks that region and interval constraints are not merely mentioned in free text but represented in usable fields where possible. iv) Provenance validation checks whether the result exposes creators, publishers, derivation relations, citation information, or responsible organisations. v) License and access validation checks whether the resource can be reused under the specified conditions. At the end four answer states are presented: supported, partially supported, unsupported, and inferred. A supported statement is backed by retrieved metadata or graph triples. A partially supported statement depends on incomplete metadata. An unsupported statement is rejected or hidden. An inferred statement is shown only if the inference method is disclosed. This design is deliberately conservative and makes answers much more useful for science. A practical example is workflow compatibility. An LLM might infer that a workflow for species occurrence data could analyse a dataset about plankton observations. The validation layer should require explicit compatibility evidence, such as declared input type, accepted format, taxonomic requirements, or documented workflow scope. Without such evidence, the result may still be suggested, but only with a clear warning that compatibility is unverified. Back in the Adriatic Sea scenario, a workflow listed with a declared input type of dcat:Distribution in Darwin Core format would pass this check. One with no declared input constraints would still be returned but flagged leaving the researcher to decide whether the missing evidence is a catalogue gap or a genuine incompatibility.

4. Evaluation methodology

We propose an evaluation protocol based on six main dimensions (see Table 1):

1. The first dimension evaluates the semantic grounding. Given a benchmark set of user requests and expert annotations, the framework is assessed on entity recognition, concept grounding, synonym handling, spatial and temporal parsing, resource-type classification, and ambiguity detection. Metrics include precision, recall, F1-score, and calibration of grounding confidence.

2. The second dimension evaluates the query-plan correctness. Experts compare generated SSPs against gold or reference plans. The assessment considers whether the plan preserves the user intent, includes necessary constraints, avoids unsupported assumptions, and represents ambiguity correctly. This is crucial because the output could include plausible resources for the wrong reasons.
3. The third dimension evaluates the retrieval quality and the validation effectiveness. For each competency question, candidate resources are judged for relevance, constraint satisfaction, metadata completeness, and usefulness for the intended scientific task. Standard information-retrieval metrics such as precision at k, recall at k, nDCG, and mean reciprocal rank can be combined with domain-specific measures, such as proportion of results with reusable licenses, complete spatial-temporal metadata, or compatible workflows.
4. The fourth dimension evaluates explanation faithfulness and user utility. Explanation faithfulness measures whether generated explanations accurately reflect the executed plan and retrieved evidence. User utility can be assessed through task-based studies with biodiversity researchers, comparing baseline catalogue search, LLM-only search, and graph-grounded agentic search. Measures include time to useful results, number of reformulations, perceived trust, ability to detect unsuitable resources, and success in assembling a reusable research bundle.

Table 1
Evaluation dimensions and candidate metrics.

Dimension	What is measured	Candidate metrics
Semantic grounding	Entity/concept linking, spatial-temporal parsing, ambiguity detection	Precision, recall, F1, confidence calibration
Query-plan correctness	Completeness and faithfulness of Semantic Search Plans	Expert rating, constraint preservation, unsupported-assumption count
Retrieval relevance	Quality of candidate resources and bundles	P@k, R@k, nDCG, MRR, metadata completeness
Validation effectiveness	Detection of missing or incompatible metadata	True/false validation outcomes, error taxonomy
Explanation faithfulness	Alignment between explanation, evidence, and executed plan	Citation/evidence accuracy, unsupported claim rate
User utility	Usefulness for biodiversity researchers	Time to useful result, reformulations, perceived trust, task success

The benchmark should include both positive and adversarial cases. Positive cases test normal discovery tasks. Adversarial cases include ambiguous species names, common names with multiple taxa, spatial names with multiple geometries, missing metadata, resources with incompatible licenses, outdated workflow descriptions, and requests that cannot be satisfied by the infrastructure. Failure-aware evaluation is important because a trustworthy system must know when to return no result, ask for clarification, or expose uncertainty. The Adriatic Sea plankton request serves as the primary positive case in our benchmark: a well-formed, multi-constraint query the system should handle end-to-end, from slot extraction through validated retrieval to a provenance-aware explanation. Adversarial variants of the same request can replace for example “plankton” with a common name that maps to several overlapping taxa or drop the temporal constraint entirely to test whether the system surfaces ambiguity rather than silently resolving it.

5. Discussion

The framework raises several design choices. One concerns the boundary between autonomy and control. Agentic AI can chain actions across catalogues, graphs, and services, but scientific infrastructures should avoid opaque automation. Our proposal favours bounded agency: the agent can plan and execute

retrieval actions, but every step is logged, every grounding is inspectable, and high-impact assumptions are surfaced to the user. A second choice concerns the role of vector retrieval. Dense retrieval is useful for matching natural-language requests with textual descriptions, but it should be complemented by graph constraints when the user asks for exact spatial, temporal, license, provenance, or service-compatibility conditions. Vector similarity can find candidates, graph validation can decide whether they satisfy the scientific contract. A third choice concerns interoperability across infrastructures. Although this paper is framed around LifeWatch ERIC, the pattern is reusable across environmental and life-science infrastructures that expose catalogues, controlled vocabularies, and service registries. The SSP could be aligned with DCAT profiles, FAIR Digital Object metadata, Research Object descriptions, or domain-specific ontology patterns. This would allow agentic search to operate across federated catalogues without reducing everything to brittle keyword matching. Finally, the framework suggests a social as well as technical role for LLMs. Many users do not know which vocabulary, metadata field, or service name expresses their need. A graph-grounded agent can act as a semantic mediator, helping users formulate better questions while preserving the accountability of catalogue-based discovery. The result is not a chatbot sprinkled over an infrastructure, but it's rather a conversation with the infrastructure through its semantics.

6. Conclusion

This paper presented a graph-grounded agentic framework for complex structured search in LifeWatch ERIC. The main idea is to use LLMs as query-planning agents whose actions are grounded in Semantic Web standards, controlled vocabularies, metadata schemas, knowledge graphs, and validation constraints. The proposed framework decomposes natural-language scientific requests into semantic search plans, grounds plan elements, executes structured retrieval, validates candidate resources, and returns provenance-aware explanations. The motivating scenario, expressed in the form of an informative need, of a researcher tracing shifts in plankton community composition across the Adriatic Sea runs as a concrete thread through every stage of this framework: slot extraction and vocabulary grounding in the SSP, SPARQL-based retrieval over catalogue knowledge-graph layers, SHACL validation of candidates, and a provenance-aware explanation the researcher can inspect and trust. The contribution is a reusable design pattern for trustworthy LLM-assisted discovery in biodiversity e-infrastructures. A productive discussion with the community will help to make stable and consolidated the main concepts and steps of the framework so that we can implement it as a sort of testbed in the LifeWatch Italy Semantic Platform (https://semantics.lifewatchitaly.eu), which already has a well-defined and structured ontological model behind that is mainly dedicated on the dataset digital object. Future work will instantiate the framework over the LifeWatch ERIC Metadata Catalogue, building a benchmark of biodiversity search competency questions, and evaluating the approach with domain experts across a wider range of digital objects (datasets, services, workflows, VREs, etc.).

Declaration on Generative AI

During the preparation of this work, the authors used ChatGPT for text polishing and rephrasing. After using this tool, the author(s) reviewed and edited the content as needed and take full responsibility for the publication's content.

References

[1] M. D. Wilkinson, M. Dumontier, I. J. Aalbersberg, et al., The fair guiding principles for scientific data management and stewardship, Scientific Data 3 (2016) 160018. doi:10.1038/sdata.2016.18.

[2] L. Vaira, E. Tzafesta, M. Fiaz, H. Ben Salah, A. Basset, The lifewatch eric metadata catalogue: A fair gateway to biodiversity and ecosystem research, Ecological Informatics 96 (2026) 103853. doi:10.1016/j.ecoinf.2026.103853.

[3] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, P. Fung, Survey of hallucination in natural language generation, ACM Computing Surveys 55 (2023) 1–38. doi:10.1145/3571730.

[4] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, et al., Language models are few-shot learners, in: Advances in Neural Information Processing Systems, volume 33, 2020, pp. 1877–1901. doi:10.48550/arXiv.2005.14165.

[5] M. T. Khalid, A.-P. Witmer, Prompt engineering for large language model-assisted inductive thematic analysis, Social Science Computer Review 44 (2026) 442–465. doi:10.1177/08944393251388098.

[6] World Wide Web Consortium, Rdf 1.1 concepts and abstract syntax, 2014. URL: https://www.w3.org/TR/rdf11-concepts/, w3C Recommendation.

[7] World Wide Web Consortium, Sparql 1.1 overview and sparql 1.1 query language, 2013. URL: https://www.w3.org/TR/sparql11-query/, w3C Recommendation.

[8] World Wide Web Consortium, Data catalog vocabulary (dcat) – version 3, 2024. URL: https://www.w3.org/TR/vocab-dcat-3/, w3C Recommendation.

[9] World Wide Web Consortium, Skos simple knowledge organization system reference, 2009. URL: https://www.w3.org/TR/skos-reference/, w3C Recommendation.

[10] World Wide Web Consortium, Prov-o: The prov ontology, 2013. URL: https://www.w3.org/TR/prov-o/, w3C Recommendation.

[11] Open Geospatial Consortium, Ogc geosparql standard: Standard for representing and querying geospatial data on the semantic web, 2024. URL: https://docs.ogc.org/is/22-047r1/22-047r1.html.

[12] World Wide Web Consortium, Semantic sensor network ontology, 2017. URL: https://www.w3.org/TR/vocab-ssn/, w3C Recommendation.

[13] World Wide Web Consortium, Shapes constraint language (shacl), 2017. URL: https://www.w3.org/TR/shacl/, w3C Recommendation.

[14] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, D. Kiela, Retrieval-augmented generation for knowledge-intensive nlp tasks, in: Advances in Neural Information Processing Systems, volume 33, 2020, pp. 9459–9474.