

SEMANTIC INTEROPERABILITY ENGINEERING REPORT: OGC ENVIRONMENTAL DATA RECORDS (EDR) AND COMMON CORE ONTOLOGIES (CCO)

ENGINEERING REPORT

DRAFT

Submission Date: 2025-03-31

Approval Date: 2025-03-31

Publication Date: 2025-05-21

Editor: Surya S. Durbha

Notice: This document is not an OGC Standard. This document is an OGC Public Engineering Report created as a deliverable in an OGC Interoperability Initiative and is *not an official position* of the OGC membership. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an OGC Standard.

Further, any OGC Engineering Report should not be referenced as required or mandatory technology in procurements. However, the discussions in this document could very well lead to the definition of an OGC Standard.

License Agreement

Use of this document is subject to the license agreement at <https://www.ogc.org/license>

Copyright notice

Copyright © 2025 Open Geospatial Consortium

To obtain additional rights of use, visit <https://www.ogc.org/legal>

Note

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. The Open Geospatial Consortium shall not be held responsible for identifying any or all such patent rights.

Recipients of this document are requested to submit, with their comments, notification of any relevant patent claims or other intellectual property rights of which they may be aware that might be infringed by any implementation of the standard set forth in this document, and to provide supporting documentation.

CONTENTS

- I. BACKGROUND AND OVERVIEW v
- II. EXECUTIVE SUMMARYvi
- III. KEYWORDS vii
- IV. CONTRIBUTORSvii
- V. FUTURE OUTLOOK vii
- VI. VALUE PROPOSITIONviii
- 1. INTRODUCTION 2
 - 1.1. Aims 3
 - 1.2. Objectives3
- 2. TOPICS 5
 - 2.1. Development of Shared Ontology 5
 - 2.2. Technical Implementation6
 - 2.3. Use case to demonstrate the usefulness of the Shared ontology21
- 3. OUTLOOK 25
- 4. SECURITY, PRIVACY AND ETHICAL CONSIDERATIONS27
- BIBLIOGRAPHY29
- ANNEX A (NORMATIVE) ABBREVIATIONS/ACRONYMS 31

LIST OF TABLES

- Table 1 – Core Entities6
- Table 2 – Properties/Relationships7
- Table 3 – Spatial/Temporal Types8
- Table 4 – Data and Measurements 8
- Table 5 – Qualities and Parameters 8
- Table 6 – Bridge/Link concepts between CCO and EDR 9

LIST OF FIGURES

Figure 1 – Shared vocabulary approach for integration of data from heterogeneous sources 3

Figure 2 – Development of a shared Ontology 5

Figure 3 – UML representation of the SHACL Schema 16

Figure 4 – UML representation of the SHACL Schema 19

Figure 5 – Illustration showing the integration of the CCO and OGC EDR using shared ontology23



BACKGROUND AND OVERVIEW

OGC has a long-standing history of advancing geospatial interoperability through both syntactic and semantic approaches. Early initiatives like the Critical Infrastructure Protection Initiatives (CIPI) tackled cross-jurisdictional data integration, employing real-time semantic translation and schema crosswalks to support emergency response. This led to early implementations of interoperable services using standards like Web Feature Service (WFS). Concurrently, the Web Mapping Testbeds and FGDC Web Mapping CAP Support initiative helped define web-based geospatial service protocols. The Conformance and Interoperability Test and Evaluation (CITE) program formalized service compliance testing, while the Geospatial Information for Sustainable Development Initial Capability Pilot (GISD-ICP) addressed thematic and international data integration. More recently, OGC Testbed-20 explored the application of FAIR data principles, reinforcing open data stewardship and machine-readability. These foundational efforts continue to inform emerging strategies such as ontology alignment, knowledge graphs, and AI-enabled semantic mediation. Mapping the OGC Environmental Data Retrieval API (EDR-API) to the Common Core Ontologies (CCO), a suite of eleven mid-level ontologies built upon the ISO-standard Basic Formal Ontology (BFO) is essential for advancing semantic interoperability in the context of emergency response and disaster management. While the OGC community has developed the EDR-API standard to support data discoverability, querying, and transfer in accordance with FAIR principles, the DoD/IC utilizes the CCO and its relevant domain extensions (such as the Atmospheric Feature Ontology, Hydrographic Feature Ontology, and Sensor Ontology) to structure information and enable effective content discovery. Aligning the OGC EDR-API with the CCO suite ensures that environmental and geospatial data can be semantically integrated and interpreted across systems, thereby improving coordination, situational awareness, and timely decision-making during emergencies.

To map the OGC EDR and the CCO, it is required to align their vocabularies. OGC EDR focuses on querying environmental data, while CCO is more general and provides a formal ontology for representing domain-agnostic concepts, including events, entities, and roles.

In order to align these two representations, a more general ontology which can act as a mediator is recommended. The development of such an ontology called the shared ontology requires deeper understanding of the underlying concepts and properties of CCO and EDR. The initial mapping begins by comparing concepts/properties in both the vocabularies based on the level of their association such as BroaderThan, NarrowerThan, Similar, etc. This will enable the development of bridging links where there is no exact match between the entities in both the vocabularies. These link/bridge concepts will be further refined iteratively to represent the commonality or mediator between the corresponding concepts/properties under consideration which will ensure interoperability. These will form the core concepts and properties in the shared development in Web Ontology Language (OWL). The shared ontology thus developed is validated using SHACL (Shapes Constraint Language) shape for validation of the core classes, cardinality restrictions, data type constraints and class constraints. To demonstrate the usefulness of the shared representation, two use cases are presented.



EXECUTIVE SUMMARY

This Engineering Report presents the development of a shared ontology to enhance interoperability between the OGC Environmental Data Retrieval (EDR) API and the Common Core Ontologies (CCO), with a particular focus on geospatial and disaster management use cases. While EDR provides a syntactic standard for accessing environmental data, CCO offers a semantically rich, domain-agnostic ontology for representing entities, events, and roles. Bridging these two paradigms requires alignment of their vocabularies through a common semantic framework. To this end, a shared ontology, referred to as `ccoedr`, has been proposed to serve as an intermediary that harmonizes concepts and properties across both standards. The alignment process began by evaluating relationships between terms in EDR and CCO using semantic association metrics such as `BroaderThan`, `NarrowerThan`, and `SimilarTo`. Where direct equivalence was absent, bridging concepts were introduced to mediate between the representations. These bridge concepts were iteratively refined and formalized in the Web Ontology Language (OWL), forming the foundation of the `ccoedr` ontology. The resulting shared ontology was validated using the Shapes Constraint Language (SHACL), ensuring logical consistency through constraints on cardinality, data types, and class relationships. This validation framework supports quality assurance in semantic interoperability and ontology governance. To demonstrate the practical utility of the shared ontology, two real-world use cases were developed. These illustrate how unified semantic representation can enhance cross-system data integration, facilitate more accurate queries, and support the reuse and discovery of environmental and disaster-related data in a standardized manner. Overall, this work contributes to OGC's broader vision of enabling semantic and syntactic interoperability in geospatial systems. It aligns with ongoing community efforts to ensure that shared vocabularies, ontology mappings, and validation frameworks support FAIR (Findable, Accessible, Interoperable, and Reusable) principles. Future work will involve expanding the scope of mappings, refining alignment techniques, and supporting operational deployments in live disaster response scenarios.

OGC has a longstanding history of addressing geospatial interoperability, with foundational efforts such as the Critical Infrastructure Protection Initiatives (CIPI) in the early 2000s exploring real-time semantic translation and schema mapping for emergency response. These initiatives addressed challenges of cross-border and cross-jurisdictional data harmonization—issues that remain central to today's disaster management needs. Other notable programs, including the Web Mapping Testbeds, the Geospatial Information for Sustainable Development Initial Capability Pilot (GISD-ICP), and the CITE program, laid critical groundwork for semantic interoperability and standardization practices still in use today. This Engineering Report builds upon and extends these prior efforts by formalizing a shared ontology—`ccoedr`, that bridges syntactic and semantic standards (EDR and CCO) to enable precise and reusable environmental and disaster data integration.



KEYWORDS

The following are keywords to be used by search engines and document catalogues.

Environmental Data Records (EDR), Common Core Ontologies (CCO), Meta-ontology, Shared ontology, Mapping, SHACL, JSON-LD, Interoperability



CONTRIBUTORS

All questions regarding this document should be directed either to the editor or to the contributors.

NAME	ORGANIZATION	ROLE
Surya S. Durbha	IIT Bombay	Editor
Gérald Fenoy	GeoLabs	Contributor
Rajat Shinde	UAH	Contributor
Ujwala Bharambe	Thadomal Shahani Engineering College	Contributor
Chetan Mahajan	IIT Bombay	Contributor



FUTURE OUTLOOK

The future outlook for integrating Common Core Ontologies (CCO) and OGC Environmental Data Retrieval (EDR) standards involves refining and expanding the developed shared ontology (ccoedr). Building on the foundation laid by the ccoedr ontology, upcoming efforts should prioritize the development of automated ontology alignment tools, refinement of Shapes Constraint Language (SHACL) validation constraints, and integration with evolving standards in linked data and semantic web technologies, while focusing on extending the mapping to encompass broader elements within CCO and potentially other OGC API standards relevant to disaster management.

Further work could involve developing more sophisticated validation techniques, exploring automated or semi-automated approaches to generate and update mappings as the source

ontologies and standards evolve. Expanding the use cases to test the shared ontology in diverse, real-world disaster response scenarios will be crucial for demonstrating and improving its practical utility in enhancing cross-community data exchange.

AI-assisted ontology matching, broader inclusion of domain-specific vocabularies, and support for real-time data interoperability scenarios such as live disaster response coordination are also recommended. Additionally, further development of use-case driven evaluation frameworks will strengthen the alignment quality and usability of the shared ontology.

As Earth observation systems, defense applications, and civilian services increasingly rely on integrated data, the continued evolution of this ontology will be essential for improving discovery, access, and trust in heterogeneous data ecosystems.



VALUE PROPOSITION

This Engineering Report addresses the critical challenge of interoperability between semantic representations like the Common Core Ontologies (CCO) used by the DoD/IC and syntactic data retrieval standards like the OGC Environmental Data Retrieval (EDR). It presents a methodology for creating a shared ontology (*ccoedr*) to bridge these different representations.

By systematically analyzing and mapping core concepts and properties, defining necessary bridge concepts, representing the shared ontology in OWL, and providing a SHACL schema for validation, this work provides a concrete framework to enhance semantic interoperability. The primary value lies in facilitating improved data discovery, integration, and flow between diverse systems and communities, ultimately supporting more effective use of geospatial data for disaster resilience and emergency management.

The *ccoedr* ontology empowers users—including governmental agencies, NGOs, and research organizations—to perform unified queries, integrate disparate datasets, and enhance data discoverability using standard languages like RDF, OWL, SHACL, and SPARQL. This approach not only supports FAIR (Findable, Accessible, Interoperable, and Reusable) principles but also enables scalable, future-ready solutions for cross-domain data fusion, particularly in scenarios such as disaster monitoring, environmental analysis, and defense intelligence.



1

INTRODUCTION

Geospatial data and information derived from commercial, civilian government, and NGO sources are increasingly used by entities within the Defense/IC community, including in their activities in support of natural disaster response and emergency management. Conversely, declassified data and information, including imagery, are used by commercial, civilian government, and NGOs for their applications in support of disaster resilience. Recent developments such as the US National Unclassified Data Lake underscore the importance of these cross-community efforts. To facilitate the flow of information between the systems used by commercial, civilian government, and NGO applications and those used by DoD/IC organizations for emergency and disaster management, interoperability between ontologies used to structure information relationships within DoD/IC information infrastructures and metadata supplied by current data retrieval processes, including standards-conformant APIs, by civilian and commercial communities should be maximized.

Earth Observation (EO) data integration is broadly dependent on the resolution of conflicts arising from the following [1] [6]:

- Data sets stemming from the same data-source with unequal updating periods;
- Data sets represented in the same data-model, but acquired by different operators;
- Data sets which are stored in similar, but not identical, data models;
- Data sets from heterogeneous sources (across geographical boundaries), which differ in data-modeling, scale, thematic content, contexts, and meaning.
- Data sets that are influenced by socio-political and cultural backgrounds.
- The resolution of such conflicts depends on the reconciliation of both syntactic and semantic heterogeneities in the data.

The classes that are used in various information sources may vary widely in their semantic representations. For example, an NGO involved in disaster response activities might capture information using the semantic enriched common core ontologies (CCO) [2] while another organization could have used a syntactic standard as proposed by OGC such as the environmental Data Retrieval (EDR) [4] for their purpose. Hence, to interoperate between these two entities and facilitate the integration of these representations, a separate, more expressive terminology is required. The semantics of this terminology may be specified by an ontology, which is then called a shared ontology or reference ontology. The semantics of the reference ontology may be specified by a standard, which is then often called meta-standard [5] (figure 1)..

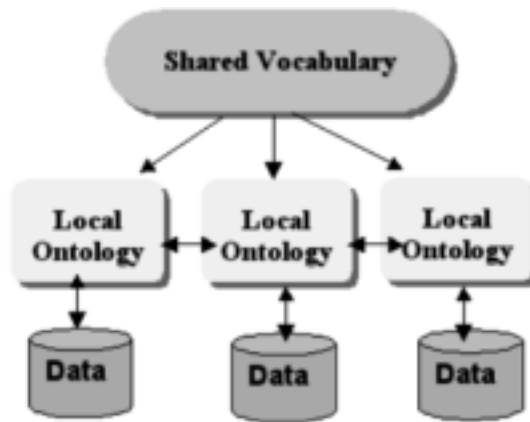


Figure 1 — Shared vocabulary approach for integration of data from heterogeneous sources

1.1. Aims

The DoD/IC uses the Common Core ontology and extensions relevant to emergency and disaster management, including the Atmospheric Feature Ontology, Hydrographic Feature Ontology, and Sensor Ontology to structure their information and facilitate discovery of relevant content. The OGC community has developed API standards, e.g. the OGC API-Features standard and the OGC EDR-API standard, to improve the interoperability of services and applications, including those relevant to emergency and disaster management, and facilitate FAIRness including data discoverability, query, and transfer.

1.2. Objectives

- Approach for developing a shared vocabulary representation methods and specific metrics for the evaluation of the cross walk mappings various metrics will be used to assess the depth of interoperability achieved.
- Methods and specific metrics for the evaluation of the cross walk mappings various metrics will be used to assess the depth of interoperability achieved.
- Gap analysis based on expressivity of concepts that is required in disaster response and management activities and that which is possible using CCO and OGC APIs.
- Identify elements that are not interoperable and elements that are ambiguous, i.e., could be mapped to multiple elements of the API content model or Common Core ontology.



2 TOPICS

The CCO is a semantic representation, whereas the OGC API EDR is a syntactic representation. Figure (2) shows the development stages of the shared ontology [5] [6] that can be used as reference for the interoperable data retrieval service.

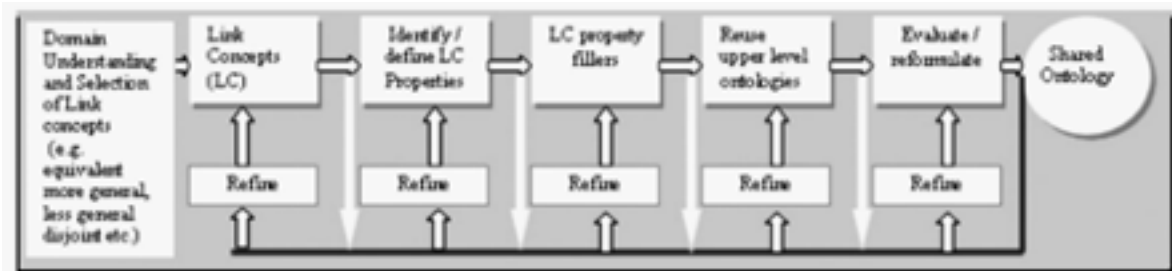


Figure 2 – Development of a shared Ontology

It begins by understanding the domain (e.g. disaster response and management) and selecting link concepts between two representations. In our case the concepts in the CCO ontology and the OGC API such as EDR. The link concepts could be any of the following: equivalent, one concept can be more or less general than the other, the concepts may be inverse or disjoint. Further, the analysis of the link concepts is required in terms of their both object properties as well as data type properties and other related information such as cardinality etc.

Based on the granularity of the concepts in each of these domain representations, we will also need to understand 1:1 (Exact) 1:M (approximate) M:M etc correspondences.. This is an interactive process and requires refinement and a cyclic approach.

To map the OGC EDR API (Environmental Data Retrieval API) and the Common Core Ontology (CCO), we need to align their conceptual and semantic frameworks. OGC EDR focuses on querying environmental data, while CCO provides a formal ontology for representing domain-agnostic concepts, including events, entities, and roles.

2.1. Development of Shared Ontology

a) Understand the terminology of the representations

- OGC EDR focuses on retrieving environmental data, describing spatial-temporal metadata and access patterns.
- CCO is designed for general-purpose use across domains, emphasizing logical formalism and consistency for entities, processes, and roles.

b) Capture the similar/broader/narrower concepts and properties such as

- Environmental Entities: Include phenomena like weather, climate, water bodies, and ecological features.
- Spatio-temporal : Both the representations address time, location, and spatial geometries.
- Processes and Roles: Processes (e.g., data retrieval, environmental phenomena) and roles (e.g., data provider, data consumer).

c) Identifying the similar concepts and bridge classes and properties for alignment

- Use CCO's upper ontology structure to align with EDR concepts, ensuring compatibility.
- Define equivalence relationships (e.g., owl:equivalentClass or owl:equivalentProperty) for similar concepts.
- Introduce bridge classes or properties where concepts and properties are narrower or broader.

d) Development of Mapping Framework using RDF/OWL

e) SHACL Validation Schema

2.2. Technical Implementation

CCO provides a broader and more general ontology for modeling entities, processes, qualities, and relationships, making it suitable for semantic interoperability across domains.

EDR is narrower and focuses specifically on environmental data retrieval through APIs, with concepts.

Table 1 provides a mapping between CCO and OGC EDR showing the BroderThan, NarrowerThan and similar concepts.

2.2.1. Cross-Walk between CCO and OGC EDR

Table 1 – Core Entities

CATEGORY	CCO CONCEPT/PROPERTY	EDR CONCEPT/PROPERTY	RELATIONSHIP
Object	cco:Object	No direct correspondence in EDR	Broader in CCO
Physical Object	cco:PhysicalObject	No direct correspondence in EDR	Broader in CCO
Process	cco:Process	No direct correspondence in EDR	Broader in CCO

CATEGORY	CCO CONCEPT/PROPERTY	EDR CONCEPT/PROPERTY	RELATIONSHIP
Event	cco:Event	No direct correspondence in EDR	Broader in CCO
Quality	cco:Quality	No direct correspondence in EDR	Broader in CCO
Location	cco:Location	edr:Location	Similar
Spatial Region	cco:SpatialRegion	edr:SpatialExtent	Similar
Temporal Region	cco:TemporalRegion	edr:TemporalExtent	Similar
Parameter	cco:Parameter	edr:Parameter	Similar
Information Object	cco:InformationObject	No direct correspondence in EDR	Broader in CCO
Dataset	cco:Dataset	edr:Dataset	Similar
Measurement	cco:Measurement	No correspondence in EDR	Broader in CCO
Data Query	No correspondence in CCO	edr:DataQuery	Broader in EDR

Table 2 — Properties/Relationships

CATEGORY	CCO CONCEPT/PROPERTY	EDR CONCEPT/PROPERTY	RELATIONSHIP
hasQuality	cco:hasQuality	No direct correspondence in EDR	Broader in CCO
locatedIn	cco:locatedIn	edr:hasLocation	Similar
participatesIn	cco:participatesIn	No direct correspondence in EDR	Broader in CCO
hasParameter	cco:hasParameter	edr:hasParameter	Similar
hasPart	cco:hasPart	No direct correspondence in EDR	Broader in CCO
hasTemporalRegion	cco:hasTemporalRegion	edr:hasTemporalExtent	Similar
hasValue	cco:hasValue	edr:hasValue	Similar
hasUnit	cco:hasUnit	edr:hasUnit	Similar
hasOutput	cco:hasOutput	No direct correspondence in EDR	Broader in

Table 3 — Spatial/Temporal Types

CATEGORY	CCO CONCEPT/PROPERTY	EDR CONCEPT/PROPERTY	RELATIONSHIP
Point	cco:Point	edr:Point	Similar
Line	cco:Line	edr:Line	Similar
Polygon	cco:Polygon	edr:Polygon	Similar
Instant	cco:Instant	No direct correspondence in EDR	Broader in CCO
Interval	cco:Interval	No direct correspondence in EDR	Broader in CCO

Table 4 — Data and Measurements

CATEGORY	CCO CONCEPT/PROPERTY	EDR CONCEPT/PROPERTY	RELATIONSHIP
Information Object	cco:InformationObject	No direct correspondence in EDR	Broader in CCO
Measurement	cco:Measurement	No correspondence in EDR	Broader in CCO
Dataset	cco:Dataset	edr:Dataset	Similar
Data Query	No correspondence in CCO	edr:DataQuery	Narrower in EDR

Table 5 — Qualities and Parameters

CATEGORY	CCO CONCEPT/PROPERTY	EDR CONCEPT/PROPERTY	RELATIONSHIP
Quality	cco:Quality	No direct correspondence in EDR	Broader in CCO
Parameter	cco:Parameter	edr:Parameter	Similar

2.2.2. Link/Bridge Concepts

The Link/bridge concepts in the shared ontology are used to align and integrate the Common Core Ontologies (CCO) and the OGC Environmental Data Retrieval (EDR). These bridge concepts ensure that the two ontologies can be interoperable. Table 4.2 shows the bridge relationships between CCO and EDR.

Table 6 – Bridge/Link concepts between CCO and EDR

CATEGORIBRIDGE/LINK CONCEPT	CCO CONCEPT	OGC EDR CONCEPT
EnvironmentalData	cco:InformationContentEntity	edr:EnvironmentalData
Sensor	cco:Device	edr:Sensor
Observation	cco:Process	edr:Observation
FeatureOfInterest	cco:Object	edr:FeatureOfInterest
Measurement	cco:QualityValue	edr:Measurement
Location	cco:Location	edr:Location
Time	cco:TemporalLocation	edr:Time
hasLocation	cco:hasSpatialLocation	edr:hasLocation
hasTime	cco:hasTemporalLocation	edr:hasTime
observes	cco:performs	edr:observes

Using the bridge concepts and the concept mappings described above the shared ontology is shown in the next section.

2.2.3. Shared Ontology (ccoedr)

The Shared ontology or reference ontology (ccoedr) serves as the top level ontology that has more expressive terminology. Below is the shared ontology representation in OWL.

```
<?xml version="1.0"?>
<rdf:RDF xmlns="http://www.w3.org/2002/07/owl#"
  xml:base="http://www.w3.org/2002/07/owl"
  xmlns:cco="http://www.ontologyrepository.com/CommonCoreOntologies#"
  xmlns:edr="http://www.opengis.net/ont/edr#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:ccoedr="http://www.example.org/ccodr#">
<Ontology/>

<!--
////////////////////////////////////
////////////////////////////////////
//
// Object Properties
```

```

//
////////////////////////////////////
////////////////////////////////////
-->

<!-- http://www.example.org/ccodr#hasLocation -->
<ObjectProperty rdf:about="http://www.example.org/ccodr#hasLocation">
  <rdfs:subPropertyOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#locatedIn"/>
  <rdfs:subPropertyOf rdf:resource="http://www.opengis.net/ont/ccodr#hasLocation"/>
</ObjectProperty>

<!-- http://www.example.org/ccodr#hasOutput -->
<ObjectProperty rdf:about="http://www.example.org/ccodr#hasOutput">
  <rdfs:subPropertyOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#hasOutput"/>
  <inverseOf rdf:resource="http://www.example.org/ccodr#retrieves"/>
  <propertyChainAxiom rdf:parseType="Collection">
    <rdf:Description rdf:about="http://www.example.org/ccodr#hasOutput"/>
  >
    <rdf:Description rdf:about="http://www.example.org/ccodr#retrieves"/>
  >
  </propertyChainAxiom>
</ObjectProperty>

<!-- http://www.example.org/ccodr#hasParameter -->

<ObjectProperty rdf:about="http://www.example.org/ccodr#hasParameter">
  <rdfs:subPropertyOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#hasParameter"/>
  <rdfs:subPropertyOf rdf:resource="http://www.opengis.net/ont/ccodr#hasParameter"/>
  <inverseOf rdf:resource="http://www.example.org/ccodr#measures"/>
  <propertyChainAxiom rdf:parseType="Collection">
    <rdf:Description rdf:about="http://www.example.org/ccodr#hasParameter"/>
    <rdf:Description rdf:about="http://www.example.org/ccodr#measures"/>
  </propertyChainAxiom>
</ObjectProperty>

<!-- http://www.example.org/ccodr#hasQuality -->
<ObjectProperty rdf:about="http://www.example.org/ccodr#hasQuality">
  <rdfs:subPropertyOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#hasQuality"/>
</ObjectProperty>

<!-- http://www.example.org/ccodr#hasTemporalExtent -->

<ObjectProperty rdf:about="http://www.example.org/ccodr#hasTemporalExtent">
  <rdfs:subPropertyOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#hasTemporalRegion"/>
  <rdfs:subPropertyOf rdf:resource="http://www.opengis.net/ont/ccodr#hasTemporalExtent"/>
</ObjectProperty>

<!-- http://www.example.org/ccodr#hasUnit -->

<ObjectProperty rdf:about="http://www.example.org/ccodr#hasUnit">
  <rdfs:subPropertyOf rdf:resource="http://www.opengis.net/ont/ccodr#hasUnit"/>
</ObjectProperty>

<!-- http://www.example.org/ccodr#hasValue -->

```

```

    <ObjectProperty rdf:about="http://www.example.org/ccedr#hasValue">
      <rdfs:subPropertyOf rdf:resource="http://www.opengis.net/ont/
edr#hasValue"/>
    </ObjectProperty>

    <!-- http://www.example.org/ccedr#measures -->

    <ObjectProperty rdf:about="http://www.example.org/ccedr#measures"/>

    <!-- http://www.example.org/ccedr#participatesIn -->

    <ObjectProperty rdf:about="http://www.example.org/ccedr#participatesIn">
      <rdfs:subPropertyOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#participatesIn"/>
    </ObjectProperty>

    <!-- http://www.example.org/ccedr#retrieves -->

    <ObjectProperty rdf:about="http://www.example.org/ccedr#retrieves"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#hasOutput -->
    <ObjectProperty rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#hasOutput"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#hasParameter -->

    <ObjectProperty rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#hasParameter"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#hasQuality -->

    <ObjectProperty rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#hasQuality"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#hasTemporalRegion -->

    <ObjectProperty rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#hasTemporalRegion"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#locatedIn -->

    <ObjectProperty rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#locatedIn"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#participatesIn -->

    <ObjectProperty rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#participatesIn"/>

    <!-- http://www.opengis.net/ont/edr#hasLocation -->

    <ObjectProperty rdf:about="http://www.opengis.net/ont/edr#hasLocation"/>

    <!-- http://www.opengis.net/ont/edr#hasParameter -->

    <ObjectProperty rdf:about="http://www.opengis.net/ont/edr#hasParameter"/>

    <!-- http://www.opengis.net/ont/edr#hasTemporalExtent -->

```



```

    <Class rdf:about="http://www.example.org/ccedr#Location">
      <rdfs:subClassOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#Location"/>
      <rdfs:subClassOf rdf:resource="http://www.opengis.net/ont/edr#Location"/>
    </Class>

    <!-- http://www.example.org/ccedr#Measurement -->

    <Class rdf:about="http://www.example.org/ccedr#Measurement">
      <rdfs:subClassOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#Measurement"/>
    </Class>

    <!-- http://www.example.org/ccedr#Parameter -->

    <Class rdf:about="http://www.example.org/ccedr#Parameter">
      <rdfs:subClassOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#Parameter"/>
      <rdfs:subClassOf rdf:resource="http://www.opengis.net/ont/edr#Parameter"/>
    </Class>

    <!-- http://www.example.org/ccedr#Point -->

    <Class rdf:about="http://www.example.org/ccedr#Point">
      <rdfs:subClassOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#Point"/>
      <rdfs:subClassOf rdf:resource="http://www.opengis.net/ont/edr#Point"/>
    </Class>

    <!-- http://www.example.org/ccedr#Polygon -->

    <Class rdf:about="http://www.example.org/ccedr#Polygon">
      <rdfs:subClassOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#Polygon"/>
      <rdfs:subClassOf rdf:resource="http://www.opengis.net/ont/edr#Polygon"/>
    </Class>

    <!-- http://www.example.org/ccedr#Process -->

    <Class rdf:about="http://www.example.org/ccedr#Process">
      <rdfs:subClassOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#Process"/>
    </Class>

    <!-- http://www.example.org/ccedr#Quality -->

    <Class rdf:about="http://www.example.org/ccedr#Quality">
      <rdfs:subClassOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#Quality"/>
    </Class>

    <!-- http://www.example.org/ccedr#SpatialRegion -->

    <Class rdf:about="http://www.example.org/ccedr#SpatialRegion">
      <rdfs:subClassOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#SpatialRegion"/>
      <rdfs:subClassOf rdf:resource="http://www.opengis.net/ont/edr#SpatialExtent"/>
    </Class>

    <!-- http://www.example.org/ccedr#TemporalRegion -->

```

```

    <Class rdf:about="http://www.example.org/ccodr#TemporalRegion">
      <rdfs:subClassOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies#TemporalRegion"/>
      <rdfs:subClassOf rdf:resource="http://www.opengis.net/ont/ccodr#TemporalExtent"/>
    </Class>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#Dataset -->

    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Dataset"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#Event -->

    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Event"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#Instant -->

    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Instant"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#Interval -->

    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Interval"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#Line -->

    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Line"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#Location -->

    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Location"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#Measurement -->

    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Measurement"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#Parameter -->

    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Parameter"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#Point -->

    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Point"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#Polygon -->

    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Polygon"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#Process -->

    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Process"/>

```

```

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#Quality -->
    <Class rdf:about="http://www.ontologyrepository.com/
CommonCoreOntologies#Quality"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#SpatialRegion -->
    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Spat
ialRegion"/>

    <!-- http://www.ontologyrepository.com/CommonCoreOntologies#TemporalRegion --
>
    <Class rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies#Temp
oralRegion"/>

    <!-- http://www.opengis.net/ont/edr#DataQuery -->
    <Class rdf:about="http://www.opengis.net/ont/edr#DataQuery"/>

    <!-- http://www.opengis.net/ont/edr#Dataset -->
    <Class rdf:about="http://www.opengis.net/ont/edr#Dataset"/>

    <!-- http://www.opengis.net/ont/edr#Line -->
    <Class rdf:about="http://www.opengis.net/ont/edr#Line"/>

    <!-- http://www.opengis.net/ont/edr#Location -->
    <Class rdf:about="http://www.opengis.net/ont/edr#Location"/>

    <!-- http://www.opengis.net/ont/edr#Parameter -->
    <Class rdf:about="http://www.opengis.net/ont/edr#Parameter"/>

    <!-- http://www.opengis.net/ont/edr#Point -->
    <Class rdf:about="http://www.opengis.net/ont/edr#Point"/>

    <!-- http://www.opengis.net/ont/edr#Polygon -->
    <Class rdf:about="http://www.opengis.net/ont/edr#Polygon"/>

    <!-- http://www.opengis.net/ont/edr#SpatialExtent -->
    <Class rdf:about="http://www.opengis.net/ont/edr#SpatialExtent"/>

    <!-- http://www.opengis.net/ont/edr#TemporalExtent -->
    <Class rdf:about="http://www.opengis.net/ont/edr#TemporalExtent"/>
</rdf:RDF>

```

Listing 1 — Shared ontology representation in OWL

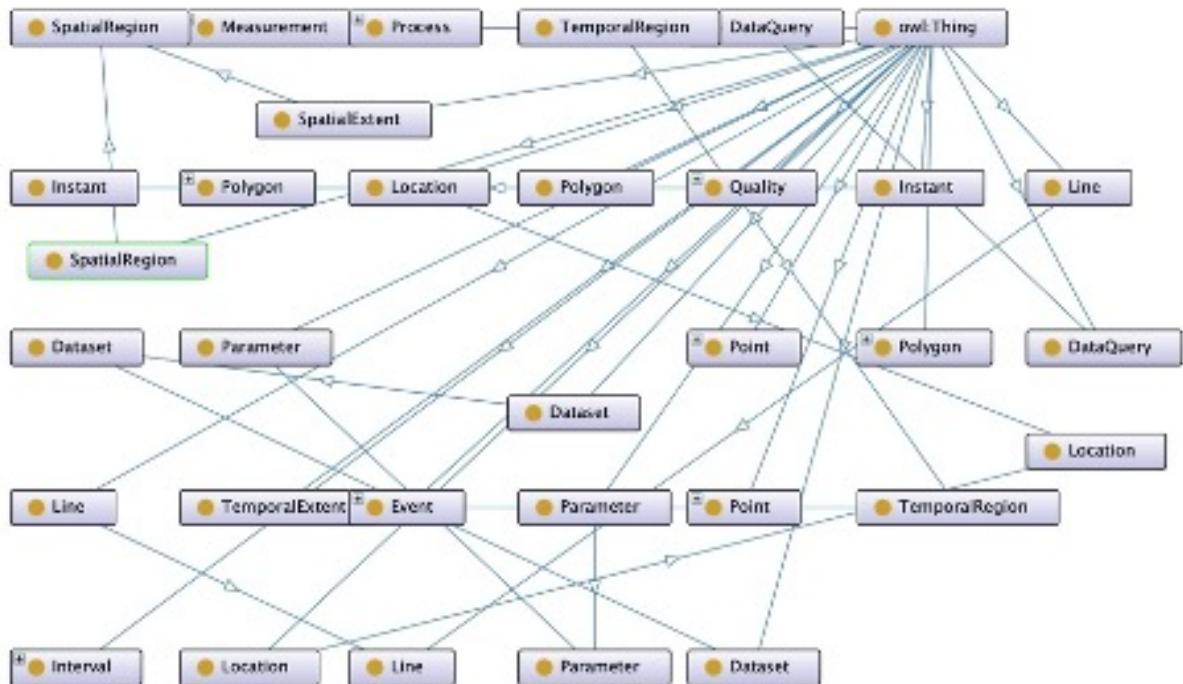


Figure 3 – UML representation of the SHACL Schema

2.2.4. SHACL Schema for the Shared Ontology

```

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sh: <http://www.w3.org/ns/shacl#> .
@prefix cco: <http://www.ontologyrepository.com/CommonCoreOntologies/> .
@prefix edr: <http://www.opengis.net/ont/edr#> .
@prefix ccoedr: <http://www.example.org/ccoedr#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

```

```

# Shape for Location
ccoedr:LocationShape
  a sh:NodeShape ;
  sh:targetClass ccoedr:Location ;
  sh:property [
    sh:path rdfs:label ;
    sh:datatype xsd:string ;
    sh:minCount 1 ;
    sh:maxCount 1 ;
  ] ;
  sh:property [
    sh:path ccoedr:SpatialRegion ;
    sh:class ccoedr:SpatialRegion ;
    sh:minCount 1 ;
    sh:maxCount 1 ;
  ] .

```

```

# Shape for Spatial Region
ccoedr:SpatialRegionShape

```

```

    a sh:NodeShape ;
    sh:targetClass ccoedr:SpatialRegion ;
    sh:property [
        sh:path rdfs:label ;
        sh:datatype xsd:string ;
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] .

# Shape for Temporal Region
ccoedr:TemporalRegionShape
    a sh:NodeShape ;
    sh:targetClass ccoedr:TemporalRegion ;
    sh:property [
        sh:path rdfs:label ;
        sh:datatype xsd:string ;
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [
        sh:path ccoedr:hasStartTime ;
        sh:datatype xsd:dateTime ;
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [
        sh:path ccoedr:hasEndTime ;
        sh:datatype xsd:dateTime ;
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] .

# Shape for Parameter
ccoedr:ParameterShape
    a sh:NodeShape ;
    sh:targetClass ccoedr:Parameter ;
    sh:property [
        sh:path rdfs:label ;
        sh:datatype xsd:string ;
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [
        sh:path ccoedr:hasUnit ;
        sh:datatype xsd:string ;
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] .

# Shape for Dataset
ccoedr:DatasetShape
    a sh:NodeShape ;
    sh:targetClass ccoedr:Dataset ;
    sh:property [
        sh:path rdfs:label ;
        sh:datatype xsd:string ;
        sh:minCount 1 ;
        sh:maxCount 1 ;
    ] ;
    sh:property [
        sh:path ccoedr:hasParameter ;
        sh:class ccoedr:Parameter ;
        sh:minCount 1 ;
    ] .

```

```

    ] ;
    sh:property [
      sh:path ccoedr:hasTemporalExtent ;
      sh:class ccoedr:TemporalRegion ;
      sh:minCount 1 ;
      sh:maxCount 1 ;
    ] ;
    sh:property [
      sh:path ccoedr:hasLocation ;
      sh:class ccoedr:Location ;
      sh:minCount 1 ;
      sh:maxCount 1 ;
    ] .

# Shape for Process
ccoedr:ProcessShape
  a sh:NodeShape ;
  sh:targetClass ccoedr:Process ;
  sh:property [
    sh:path rdfs:label ;
    sh:datatype xsd:string ;
    sh:minCount 1 ;
    sh:maxCount 1 ;
  ] ;
  sh:property [
    sh:path ccoedr:hasLocation ;
    sh:class ccoedr:Location ;
    sh:minCount 1 ;
    sh:maxCount 1 ;
  ] ;
  sh:property [
    sh:path ccoedr:hasTemporalExtent ;
    sh:class ccoedr:TemporalRegion ;
    sh:minCount 1 ;
    sh:maxCount 1 ;
  ] ;
  sh:property [
    sh:path ccoedr:hasOutput ;
    sh:class ccoedr:Dataset ;
    sh:minCount 1 ;
  ] .

# Shape for Sensor (Physical Object)
ccoedr:SensorShape
  a sh:NodeShape ;
  sh:targetClass ccoedr:PhysicalObject ;
  sh:property [
    sh:path rdfs:label ;
    sh:datatype xsd:string ;
    sh:minCount 1 ;
    sh:maxCount 1 ;
  ] ;
  sh:property [
    sh:path ccoedr:locatedIn ;
    sh:class ccoedr:Location ;
    sh:minCount 1 ;
    sh:maxCount 1 ;
  ] ;
  sh:property [
    sh:path ccoedr:measures ;
    sh:class ccoedr:Parameter ;
    sh:minCount 1 ;
  ] .

```

```

# Shape for Data Query
ccoedr:DataQueryShape
  a sh:NodeShape ;
  sh:targetClass ccoedr:DataQuery ;
  sh:property [
    sh:path rdfs:label ;
    sh:datatype xsd:string ;
    sh:minCount 1 ;
    sh:maxCount 1 ;
  ] ;
  sh:property [
    sh:path ccoedr:hasParameter ;
    sh:class ccoedr:Parameter ;
    sh:minCount 1 ;
  ] ;
  sh:property [
    sh:path ccoedr:hasLocation ;
    sh:class ccoedr:Location ;
    sh:minCount 1 ;
    sh:maxCount 1 ;
  ] ;
  sh:property [
    sh:path ccoedr:hasTemporalExtent ;
    sh:class ccoedr:TemporalRegion ;
    sh:minCount 1 ;
    sh:maxCount 1 ;
  ] ;
  sh:property [
    sh:path ccoedr:retrieves ;
    sh:class ccoedr:Dataset ;
    sh:minCount 1 ;
  ] .

```

Listing 2 – SHACL Schema for the Shared Ontology

The above SHACL schema defines constraints and validation rules to ensure that the data conforms to the structure and relationships defined in the CCO-EDR shared ontology. The UML representation of the SHACL above is shown in figure.

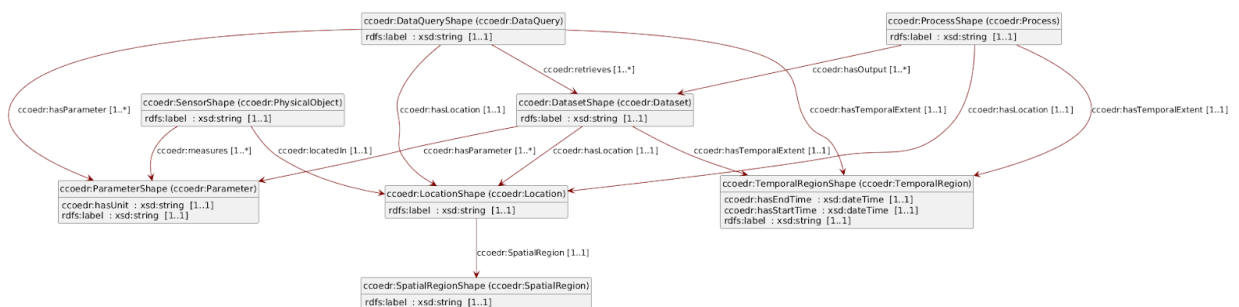


Figure 4 – UML representation of the SHACL Schema

2.2.5. JSON-LD Context Document

```

{
  "@context": {
    "id": "@id",
    "type": "@type",

```

```

"graph": "@graph",
"cco": "http://www.ontologyrepository.com/CommonCoreOntologies/",
"ccoedr": "http://www.example.org/ccodr#",
"edr": "http://www.opengis.net/ont/edr#",
"rdf": "http://www.w3.org/1999/02/22-rdf-syntax-ns#",
"rdfs": "http://www.w3.org/2000/01/rdf-schema#",
"sh": "http://www.w3.org/ns/shacl#",
"xsd": "http://www.w3.org/2001/XMLSchema#",
"DataQueryShape": "ccoedr:DataQueryShape",
"DatasetShape": "ccoedr:DatasetShape",
"FloodEventShape": "ccoedr:FloodEventShape",
"LocationShape": "ccoedr:LocationShape",
"ParameterShape": "ccoedr:ParameterShape",
"SensorShape": "ccoedr:SensorShape",
"TemporalRegionShape": "ccoedr:TemporalRegionShape",
"hasEndTime": {
  "@id": "ccoedr:hasEndTime",
  "@type": "http://www.w3.org/2001/XMLSchema#dateTime"
},
"hasLocation": {
  "@id": "ccoedr:hasLocation",
  "@type": "@id"
},
"hasOutput": {
  "@id": "ccoedr:hasOutput",
  "@type": "@id",
  "@container": "@set"
},
"hasParameter": {
  "@id": "ccoedr:hasParameter",
  "@type": "@id",
  "@container": "@set"
},
"hasStartTime": {
  "@id": "ccoedr:hasStartTime",
  "@type": "http://www.w3.org/2001/XMLSchema#dateTime"
},
"hasTemporalExtent": {
  "@id": "ccoedr:hasTemporalExtent",
  "@type": "@id"
},
"hasUnit": {
  "@id": "ccoedr:hasUnit",
  "@type": "http://www.w3.org/2001/XMLSchema#string"
},
"label": {
  "@id": "rdfs:label",
  "@type": "http://www.w3.org/2001/XMLSchema#string"
},
"locatedIn": {
  "@id": "ccoedr:locatedIn",
  "@type": "@id"
},
"measures": {
  "@id": "ccoedr:measures",
  "@type": "@id",
  "@container": "@set"
},
"retrieves": {
  "@id": "ccoedr:retrieves",
  "@type": "@id",
  "@container": "@set"
},

```

```

    "SpatialRegion": {
      "@id": "ccoedr:SpatialRegion",
      "@type": "@id"
    }
  }
}

```

Listing 3 — JSON-LD Context Document

2.3. Use case to demonstrate the usefulness of the Shared ontology

Below is a use case that demonstrate the use of the shared ontology to interoperate between the COO and OGC EDR.

2.3.1. Use Case : Flood monitoring

Below is an example data snippet in RDF that captures various details related to flood monitoring.

```

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix ccoedr: <http://www.example.org/ccoedr#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

# Flood Event
<http://example.org/flood#FloodEvent_2023>
  a ccoedr:Process ;
  rdfs:label "Flood Event 2023" ;
  ccoedr:hasLocation <http://example.org/location#RiverXYZ> ;
  ccoedr:hasTemporalExtent <http://example.org/time#Interval_2023> ;
  ccoedr:hasOutput <http://example.org/data#WaterLevelDataset> .

# Location
<http://example.org/location#RiverXYZ>
  a ccoedr:Location ;
  rdfs:label "River XYZ" ;
  ccoedr:SpatialRegion <http://example.org/spatial#Region_RiverXYZ> .

# Spatial Region
<http://example.org/spatial#Region_RiverXYZ>
  a ccoedr:SpatialRegion ;
  rdfs:label "Spatial Region of River XYZ" .

# Temporal Region
<http://example.org/time#Interval_2023>
  a ccoedr:TemporalRegion ;
  rdfs:label "October 1-5, 2023" ;
  ccoedr:hasStartTime "2023-10-01T00:00:00Z"^^xsd:dateTime ;
  ccoedr:hasEndTime "2023-10-05T23:59:59Z"^^xsd:dateTime .

# Sensor
<http://example.org/sensor#Sensor_001>
  a ccoedr:PhysicalObject ;

```

```

    rdfs:label "Water Level Sensor 001" ;
    ccoedr:locatedIn <http://example.org/location#RiverXYZ> ;
    ccoedr:measures <http://example.org/parameter#WaterLevel> .

# Parameter (Water Level)
<http://example.org/parameter#WaterLevel>
  a ccoedr:Parameter ;
  rdfs:label "Water Level" ;
  ccoedr:hasUnit "meters" .

# Dataset
<http://example.org/data#WaterLevelDataset>
  a ccoedr:Dataset ;
  rdfs:label "Water Level Dataset" ;
  ccoedr:hasParameter <http://example.org/parameter#WaterLevel> ;
  ccoedr:hasTemporalExtent <http://example.org/time#Interval_2023> ;
  ccoedr:hasLocation <http://example.org/location#RiverXYZ> .

# Data Query
<http://example.org/query#WaterLevelQuery>
  a ccoedr:DataQuery ;
  rdfs:label "Water Level Data Query" ;
  ccoedr:hasParameter <http://example.org/parameter#WaterLevel> ;
  ccoedr:hasLocation <http://example.org/location#RiverXYZ> ;
  ccoedr:hasTemporalExtent <http://example.org/time#Interval_2023> ;
  ccoedr:retrieves <http://example.org/data#WaterLevelDataset> .

```

Listing 4 – Sample RDF data

By using the shared ontology(ccoedr) the following SPARQL queries can be asked on the above example RDF data.

```

PREFIX ccoedr: <http://www.example.org/ccoedr#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?event ?eventLabel ?location ?locationLabel
WHERE {
  ?event a ccoedr:Process ;
         rdfs:label ?eventLabel ;
         ccoedr:hasLocation ?location .
  ?location rdfs:label ?locationLabel .
}

```

Listing 5 – Query 1: Retrieve all flood events with their locations

```

PREFIX ccoedr: <http://www.example.org/ccoedr#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?sensor ?sensorLabel ?parameter ?parameterLabel
WHERE {
  ?sensor a ccoedr:PhysicalObject ;
         rdfs:label ?sensorLabel ;
         ccoedr:measures ?parameter .
  ?parameter rdfs:label ?parameterLabel .
}

```

Listing 6 – Query 2: Retrieve all sensors and their measured parameters

```

PREFIX ccoedr: <http://www.example.org/ccoedr#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT ?event ?eventLabel ?startTime ?endTime

```

```

WHERE {
  ?event a ccoedr:Process ;
        rdfs:label ?eventLabel ;
        ccoedr:hasTemporalExtent ?temporalExtent .
  ?temporalExtent ccoedr:hasStartTime ?startTime ;
        ccoedr:hasEndTime ?endTime .
}

```

Listing 7 – Query 3: Retrieve all flood events with their temporal extents

```

PREFIX ccoedr: <http://www.example.org/ccoedr#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?sensor ?sensorLabel
WHERE {
  ?sensor a ccoedr:PhysicalObject ;
        rdfs:label ?sensorLabel ;
        ccoedr:locatedIn <http://example.org/location#RiverXYZ> .
}

```

Listing 8 – Query 4: Retrieve all sensors located in a specific region

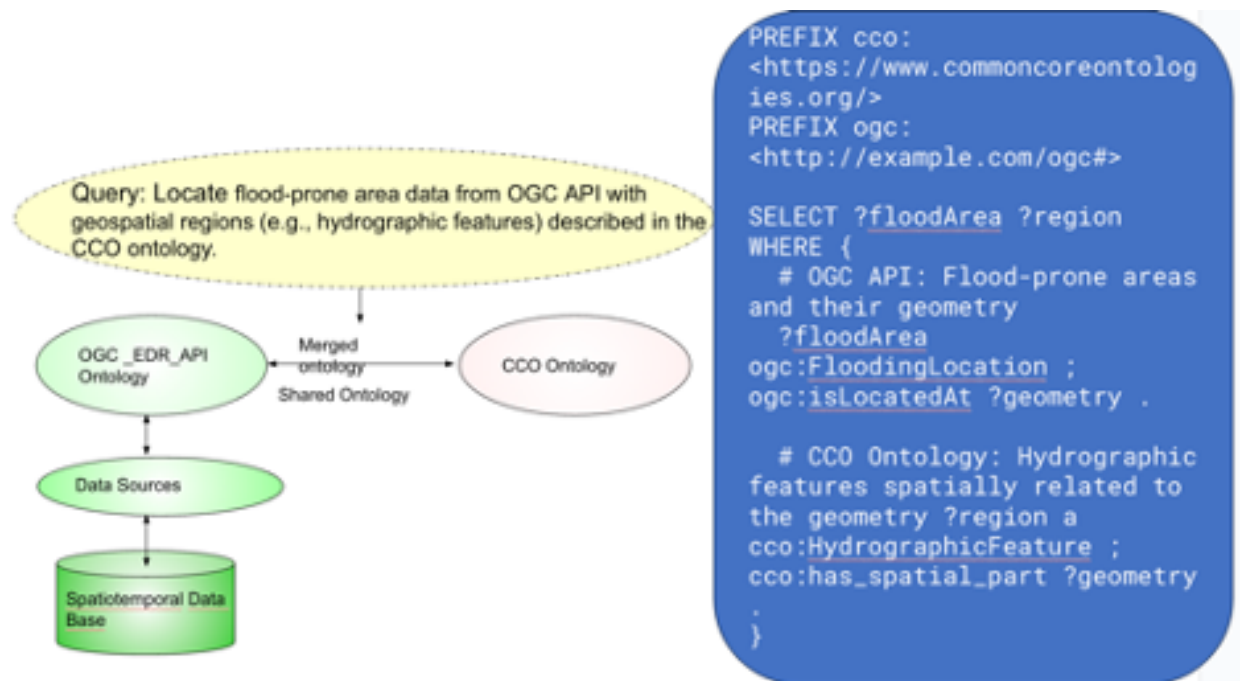


Figure 5 – Illustration showing the integration of the CCO and OGC EDR using shared ontology

3

OUTLOOK

The development of a shared ontology enables interoperability between Common Core Ontologies (CCO) and OGC EDR. The shared ontology takes into consideration various mappings between the two representations (CCO and EDR) by identifying BroaderThan, NarrowerThan, Similar, etc. relationships between the concepts in the two representations as well as the properties. Subsequently the bridge concepts between them are identified and used in the alignment of the two representations by the shared ontology, which is developed in OWL. A SHACL (Shape Constraint Language) shape is developed for validating RDF data against a set of constraints. To create a SHACL shape for the shared ontology, we define constraints for classes, properties, and individuals. Finally, the usefulness of the developed shared ontology has been demonstrated in two use cases using SPARQL.

4

SECURITY, PRIVACY AND ETHICAL CONSIDERATIONS

SECURITY, PRIVACY AND ETHICAL CONSIDERATIONS

As the integration of semantic and syntactic standards such as the Common Core Ontologies (CCO) and the OGC Environmental Data Retrieval (EDR) advances, ensuring robust security, privacy, and ethical foundations becomes paramount.

In future deployments, particularly those involving real-time disaster response and climate monitoring, stringent data protection protocols must be enforced to prevent unauthorized access and misuse. This includes implementing secure data transmission mechanisms, authentication workflows, and access controls tailored to the operational needs of diverse stakeholders.

Ethical considerations are equally crucial in maintaining trust and accountability. All AI-assisted tools used for ontology matching, data evaluation, or decision support must adhere to responsible AI and FAIR principles. Particular attention should be given to minimizing algorithmic bias, especially when shared ontologies are deployed in high-stakes domains such as emergency response.

In the future, to support transparency, the shared ontology (ccoedr) and associated mappings are planned to include provenance metadata, processing histories, and known limitations as part of the vocabulary. This facilitates reproducibility and informed decision-making while supporting FAIR data practices. Furthermore, periodic audits and community review mechanisms are encouraged to continuously monitor the ethical and technical integrity of ontology-driven systems.

By embedding these considerations into the core design and operationalization of the developed shared ontology (ccoedr), this initiative not only enhances technical capabilities but also fosters trust, accountability, and resilience in multi-stakeholder geospatial data ecosystems.



BIBLIOGRAPHY





BIBLIOGRAPHY

- [1] V. Walter and D. Fritsch, "Matching spatial data sets: Statistical approach", Int. J. Geograph. Inf. Sci., vol. 13, pp. 445-473, 1999.
- [2] Jensen, Mark, et al. "The Common Core Ontologies." arXiv preprint arXiv:2404.17758 (2024).
- [3] OGC API Features: <https://ogcapi.ogc.org/features/>
- [4] OGC API EDR: <https://ogcapi.ogc.org/edr/>
- [5] S. S. Durbha and R. L. King, Semantics-enabled framework for knowledge discovery from Earth observation data archives, IEEE Transactions on Geoscience and Remote Sensing, vol. 43, no. 11, pp. 2563-2572, Nov. 2005, doi: 10.1109/TGRS.(2005).
- [6] S.S. Durbha, R.L. King, V.P. Shah, N.H. Younan, A framework for semantic reconciliation of disparate earth observation thematic data, Computers & Geosciences, Volume 35, Issue 4, 2009, Pages 761-773. (2009)
- [7] U. Bharambe, S. S. Durbha, Adaptive Pareto-based approach for geo-ontology matching, Computers & Geosciences, Volume 119, (2018).



ANNEX A (NORMATIVE) ABBREVIATIONS/ACRONYMS



ANNEX A (NORMATIVE) ABBREVIATIONS/ACRONYMS

CCO	Common Core Ontologies
DoD	Department of Defense
EDR	Environmental Data Retrieval
NGO	Non-Governmental Organization
OWL	Web Ontology Language
RDF	Resource Description Framework
SHACL	Shapes Constraint Language
SPARQL	SPARQL Protocol and RDF Query Language