Policy-Driven Contract Negotiation using ODRL and Verifiable Credentials

Yassir Sellami[1]

[1]Gaia-X AISBL, Brussels, Belgium

Abstract

Data Space participants need to automate data access decisions across organizations without pre-established trust relationships while still meeting compliance, sovereignty, and data governance requirements. However, current implementations often leave a gap between what the policy requires (expressed as constraints) and how the Consumer can consistently select and present the right credentials in a standards-based way. We propose a concrete, interoperable approach to contract negotiation between a Consumer and a Provider in a Data Space where access to data or services is governed by an ODRL policy whose evaluated values are expressed as Verifiable Credential (VC) claims. The Provider publishes an ODRL Offer policy using the ODRL Verifiable Credential (ODRL-VC) Profile. During negotiation, the Consumer proves compliance by presenting the required credentials from a Wallet. To follow current widely adopted standards, the Provider acts as a Verifier under the OpenID for Verifiable Presentations (OpenID4VP) protocol and requests the required credentials using Digital Credentials Query Language (DCQL). We define a protocol that: (i) derives DCQL requests from ODRL-VC constraints and (ii) binds presentations to the negotiation context.

Keywords

Verifiable Credentials, Attribute-based access control, Credential Exchange, Policy evaluation, ODRL

1. Introduction

As organizations increasingly seek to exchange data across institutional boundaries, the need for structured, policy-driven access control has become paramount. Data Spaces have emerged as a promising architectural paradigm to enable such cross-organizational data sharing while preserving data sovereignty: Providers publish offers, Consumers negotiate terms, and access is granted only once a binding Data Usage Agreement is reached, given that the Provider is the Data Rights Holder [1]. The Dataspace Protocol (DSP) standardizes this life cycle, defining a Contract Negotiation state machine along with the message types exchanged by Consumers and Providers to progress from an initial request to a finalized Agreement under which data exchange becomes available [2].

Central to this negotiation life cycle is the expression of access conditions in a machine-readable and interoperable policy language. The W3C Open Digital Rights Language (ODRL) [3, 4] has been adopted as the standard for this purpose within Data Space ecosystems such as Gaia-X AISBL [5] and the Data Spaces Support Centre [6], providing a vocabulary for expressing permissions, prohibitions, and obligations over data assets. However, in practice, a significant gap remains between what a policy *requires* — expressed as constraints over attributes of the requesting party — and how a Consumer can consistently discover, select, and present the correct credentials to satisfy those constraints in a standardized, interoperable manner.

Verifiable Credentials (VCs), as defined by the W3C [7], provide a cryptographically verifiable mechanism for expressing and communicating claims about a subject. When combined with ODRL, they offer a trustworthy approach to policy evaluation: rather than relying on implicit or proprietary identity assertions, a Provider can specify precisely which VC claims must be satisfied, and a Consumer can present verifiable proof of compliance. The ODRL Verifiable Credential Profile (ODRL-VC Profile) formalizes this combination by defining how an ODRL policy can reference VC claims using JSONPath

2nd OPAL Workshop - ODRL and beyond: practical applications and challenges for policy-based access and usage control, co-located with the 23rd European Semantic Web Conference (ESWC), May 10-14, 2026 | Dubrovnik , Croatia

yassir.sellami@gaia-x.eu (Y. Sellami)

ysellami.com (Y. Sellami)

expressions and credential type identifiers [8]. This makes it possible to express fine-grained, attribute-level conditions directly within a negotiable policy Offer.

On the credential side, OpenID for Verifiable Presentation (OpenID4VP) [9] defines a standardized protocol flow in which a Wallet presents credentials to a Verifier, building on the established OpenID Connect specification. Complementing this, the Digital Credentials Query Language (DCQL) [10] provides a structured JSON-based query mechanism that allows a Verifier to request specific credentials and claims in a format-agnostic way. Together, OpenID4VP and DCQL enable a Provider to request the exact credentials required by its policy and cryptographically verify the Consumer's compliance before concluding negotiation.

Despite the existence of these complementary standards, no concrete protocol has been proposed that systematically bridges ODRL/VC policy modeling, Dataspace Protocol contract negotiation, and OpenID4VP-based credential presentation into a coherent, end-to-end flow. This paper addresses that gap.

The remainder of this paper is structured as follows. Section 2 provides background on ODRL, W3C Verifiable Credentials, the ODRL-VC Profile, OpenID4VP, and the Dataspace Protocol, establishing the conceptual foundations for the proposed approach, and relevant related work.
Section 3 defines the functional and non-functional requirements that the protocol must satisfy, derived from the constraints of cross-organizational trust and compliance in Data Spaces.
Section 4 presents the formal definition and modeling of the protocol, including the derivation of DCQL queries from ODRL-VC constraints and the binding of credential presentations to the negotiation context.
Section 5 discusses the implications of the proposed approach and its limitations.
Finally, Section 6 concludes the paper and outlines directions for future work.

2. Background & Related Work

This section summarizes the relevant standards and profiles that our protocol composes.

2.1. Open Digital Rights Language (ODRL)

ODRL (Open Digital Rights Language) is a W3C policy language for expressing permissions, prohibitions, and obligations over assets and services. Standard ODRL is expressive, but policy enforcement often depends on external evaluation[1], with multiple and diverse implementations such as: *Open Digital Rights Enforcement (ODRE)* [11], *ODRL-PAP* [12], or *Framework for ODRL Rule Compliance through Evaluation (FORCE)* [13].

2.2. Verifiable Credentials

Verifiable Credentials (VCs) are a standard way to express digitally signed claims about an entity (a person, organization, or device) that can be verified cryptographically and exchanged across systems without relying on a single central database. In the VC model, an Issuer attests claims, a Holder stores them (typically in a Wallet), and a Verifier checks their authenticity and validity, enabling interoperable trust across organizational boundaries. This makes VCs a key data sovereignty enabler in Data Spaces: participants can prove required properties while keeping control over when and with whom those proofs are shared. For privacy-preserving presentation, selective-disclosure mechanisms such as SD-JWT allow the Holder to reveal only the minimal subset of claims needed for a specific policy check, reducing unnecessary data exposure while still providing verifiable integrity [7].

[1] https://www.w3.org/community/odrl/implementations/

2.3. ODRL Verifiable Credential Profile

The ODRL VC Profile extends ODRL so that constraints can directly refer to claims in Verifiable Credentials. It is intentionally broad and not restricted to a single domain. The profile explicitly frames two parties:

- **Provider**: owns rights over the data/service and defines the policy (often an ODRL *Offer*).
- **Consumer**: seeks to use the data/service and must satisfy policy constraints using credentials.

The profile introduces terms under the `ovc` namespace, notably:

- `ovc:constraint`: a refinement of an ODRL constraint specialized for VC-based evaluation.
- `ovc:leftOperand`: uses JSONPath so the policy can reference nested claims in credential payloads without hardcoding a verifier-specific parsing engine. This makes policy portable across implementations, as long as they agree on the credential schema and context.
- `ovc:credentialSubjectType`: the expected VC type (or credential subject type) against which the constraint applies.

The profile states that an `ovc:constraint` must include `ovc:leftOperand`, `odrl:operator`, `ovc:credentialSubjectType` and `odrl:rightOperand`. This is the core bridge between ODRL policy semantics and concrete credential claims. The profile also positions the provider as policy assigner (typically publishing an ODRL *Offer*) and the consumer as the party that must satisfy the policy using credentials (policy assignee). This transforms ODRL into a machine-verifiable attribute-based access control (ABAC) policy language for Data Spaces.

2.4. OpenID4VP and DCQL

OpenID for Verifiable Presentations (OpenID4VP) standardizes how a verifier requests and receives credential presentations from a Wallet. The OpenID4VP 1.0 specification introduces a `dcql_query` request parameter for Digital Credentials Query Language (DCQL) and supports requesting credentials in multiple formats, including W3C VCs [9].

DCQL enables a verifier to request only the credentials and claims needed for a transaction. This is especially useful in our use case because ODRL policies often constrain only a subset of claims (e.g. country code, membership level, accreditation status) on any given credential.

2.5. Dataspace Protocol and Contract Negotiation

The Dataspace Protocol defines interoperable interactions for catalog publication, contract negotiation, and data transfer. In the contract negotiation part, a provider and a consumer exchange messages that drive a shared state machine. The protocol describes the roles (provider, consumer), message types (contract request, offer, agreement, agreement verification, termination), and states such as *REQUESTED*, *OFFERED*, *ACCEPTED*, *AGREED*, and *VERIFIED* [2].

3. Requirements

This section derives the requirements for a credential-governed contract negotiation protocol from four foundational pillars: *(i)* data sovereignty and governance principles as codified in the Gaia-X Trust Framework [5] and related global Data Space initiatives; *(ii)* the business and operational needs of participants in a decentralized ecosystem; *(iii)* the imperative for domain-agnostic, standards-based interoperability; and *(iv)* data privacy obligations and attribute-based access control best practices. Together, these pillars motivate both the functional capabilities the protocol must provide and the non-functional properties it must uphold.

3.1. Functional Requirements

FR1. Provider-defined policy. The Provider MUST publish an ODRL Offer that expresses rules based on claims that can be asserted by one or more Verifiable Credential.

FR2. Consumer proof obligation. The Consumer MUST present Verifiable Credentials that satisfy all mandatory policy rules.

FR3. Policy-to-credential translation. The Provider's Verifier MUST deterministically derive a set of credential claims from the set of Offer Policy constraints, without requiring external input.

FR4. Selective disclosure. The Wallet SHOULD disclose only the credentials and claims required to fulfill the policy, in conformance with the data-minimization principle.

FR5. Cryptographic verification. The Provider MUST validate and verify all claims used from credentials before proceeding to policy evaluation.

FR6. Policy evaluation. After successful cryptographic verification, the Provider MUST evaluate each policy constraint by comparing each `leftOperand` to the presented credential claims and comparing the result against `rightOperand` using the specified ODRL operator.

3.2. Non-Functional Requirements

NFR1. Interoperability. All artifacts MUST conform to published open standards, requiring no bilateral extensions.

NFR2. Privacy minimization. The requested input from the Consumer derived from an ODRL Offer MUST request no more claims than those referenced by the policy rules in the active constraints.

NFR3. Auditability. The Provider MUST retain a tamper-evident record linking the policy version, extracted constraints, credentials, and evaluation result (such as a policy compliance report [14]) for dispute resolution.

NFR4. Determinism. Any mapping rules used for evaluation and credential extraction must be deterministic: given the same Offer policy and the same credentials, any conforming implementation MUST produce an equivalent evaluation result.

NFR5. Extensibility. The protocol SHOULD support multiple policy and credential formats (VC-JWT, SD-JWT, mdoc).

4. Protocol definition

This section specifies the end-to-end negotiation protocol. We first define the actors and artifacts involved, then present the step-by-step interaction sequence, formalize the ODRL-to-DCQL mapping rules, illustrate the protocol with two policy examples, and address failure handling at the end.

4.1. Actors

- **Provider**: Provide access to dataset/service offering and publishes ODRL policy; runs or delegates policy reasoning and VC verification.
  - **Verifier**: by the provider or an authorized third-party, OpenID4VP verifier component that builds DCQL requests and verifies credentials.
  - **Policy Engine**: by the provider or an authorized third-party, it must evaluate every rule in the policy and each `ovc:constraint` expression against the presented claims

- **Consumer**: Negotiates contract and requests access to a dataset/service offering.
  - **Wallet**: by the consumer or an authorized third-party, a standards-conformant OpenID4VP Wallet [9] that holds VCs and responds to DCQL-parameterized Authorization Requests.

4.2. Artifacts

- **ODRL Offer Policy**: ODRL Policy using the `https://w3id.org/gaia-x/ovc/1/` profile.
- **Evaluation Request**: an evaluation request from the consumer including: `requestedAction`, `requestingParty` and `requestedTarget` [15].
- **OpenID4VP Authorization Request**: carries a `dcql_query`, a `nonce`, and the verifier's `client_id`.
- **OpenID4VP Authorization Response**: carries a signed `vp_token` bound to the nonce and verifier audience.
- **Contract Negotiation State (CNS)**: the current state of the contract negotiation corresponding to Dataspace Protocol Contract Negotiation states [2].

4.3. Interaction Sequence

The following sequence focuses on individual interactions that exchange VCs required by the ODRL policy. The data exchange step is omitted as it is out of scope for the protocol.

S1. Offer publication: The Provider publishes an ODRL Offer. The Offer MUST include at least one Permission rule carrying `ovc:constraint` elements and MUST declare `profile: "https://w3id.org/gaia-x/ovc/1/"` to signal ODRL-VC Profile conformance [8].

S2. Catalog discovery: The Consumer retrieves the Provider's catalog entry, and requests the target offer. The Contract Negotiation CNS enters state REQUESTED.

S3. Offer delivery: The Provider sends the full ODRL Offer to the Consumer. The CNS transitions to state OFFERED.

S4. Evaluation Request: The Consumer submits an Evaluation Request to the Provider, expressing its intent to be bound by the terms of the referenced Offer. The Evaluation Request MUST reference the Offer UID so that the Policy Engine can locate the applicable `ovc:constraint` elements. The CNS transitions to state ACCEPTED.

S5. Constraint extraction and normalization: The Provider's Policy Engine parses the ODRL Offer and extracts all `ovc:constraint` elements from applicable Permission rules.

$$\langle\, \mathit{credentialSubjectType},\ \mathit{leftOperand},\ \mathit{operator},\ \mathit{rightOperand} \,\rangle$$

S6. DCQL derivation: The Provider's Verifier constructs a `dcql_query` following the mapping rules defined in 4.4. Constraints that target the same credential type are grouped into a single credential query object where possible, minimizing the number of credentials the Consumer must present (privacy minimization requirement NFR2).

S7. OpenID4VP Authorization Request generation: The Provider's Verifier creates an OpenID4VP Authorization Request containing the `dcql_query`, a cryptographically random `nonce` bound to the CNS transaction identifier, and the verifier's `client_id`.

S8. Request delivery: The Provider's Verifier forwards the OpenID4VP Authorization Request to the Consumer as part of the response to the Evaluation Request. The CNS remains in state ACCEPTED while awaiting the credential response.

S9. Wallet presentation flow and user consent: The Consumer's Wallet matches the `dcql_query` against locally held credentials, filtering by format and credential type. The Wallet presents the matched credentials and the claims to be disclosed to the user for review. If no matching credential is found, the Wallet returns an error; the Consumer MUST propagate this failure to the Provider, which MUST terminate the CNS. Upon explicit user consent, the Wallet proceeds to construct the Verifiable Presentation; otherwise, the CNS is terminated.

S10. VP construction and response: The Wallet assembles a Verifiable Presentation containing the disclosed credentials, applying selective disclosure where the credential format supports it [16]. The Wallet then signs it with the holder key and returns it to the Verifier in the OpenID4VP Authorization Response (`vp_token`)

S11. Cryptographic validation and verification: The Provider's Verifier performs the following checks in order:

(i) nonce and audience binding;

(ii) holder proof verification;

(iii) per-credential signature or proof verification;

(iv) credential status check (revocation or suspension) where a status mechanism is present [17];

(v) issuance date and expiration date validity.

A failure at any step MUST cause the Provider to classify the Evaluation Request as *invalid* and MUST terminate the CNS.

S12. ODRL Policy evaluation: For each normalized constraint tuple, the Provider's Policy Engine applies the JSONPath expression in `ovc:leftOperand` [18] to the relevant `credentialSubject`, extracts the claim value, and evaluates the ODRL operator against the `rightOperand`. All mandatory constraints MUST evaluate to `true` for the Evaluation Request to be classified as *valid*.

- **[Valid evaluation]:** The Provider classifies the Evaluation Request as valid. The CNS advances to state AGREED.
- **[Invalid evaluation]:** The Provider classifies the Evaluation Request as invalid. The Provider issues a message identifying the unsatisfied constraint operands. The CNS transitions to state TERMINATED.

S13. Negotiation progression: On a passing evaluation, the Provider reports success to the Consumer and sends an Agreement. The Consumer acknowledges the message and confirms the Agreement, advancing the CNS to state VERIFIED. The Provider MUST persist an evaluation record at this point (NFR3).

S14. Data or service access: The Provider enables the data transfer endpoint until completion, at which point the contract negotiation (CNS) transitions to the FINALIZED state.

Sequence diagram. The following sequence diagram illustrates the nominal end-to-end flow from catalog discovery to contract finalization (steps S1–S14), encompassing ODRL-VC policy evaluation, OpenID4VP/DCQL-based credential presentation, and DSP contract negotiation state transitions. Failure and termination paths are omitted for clarity.

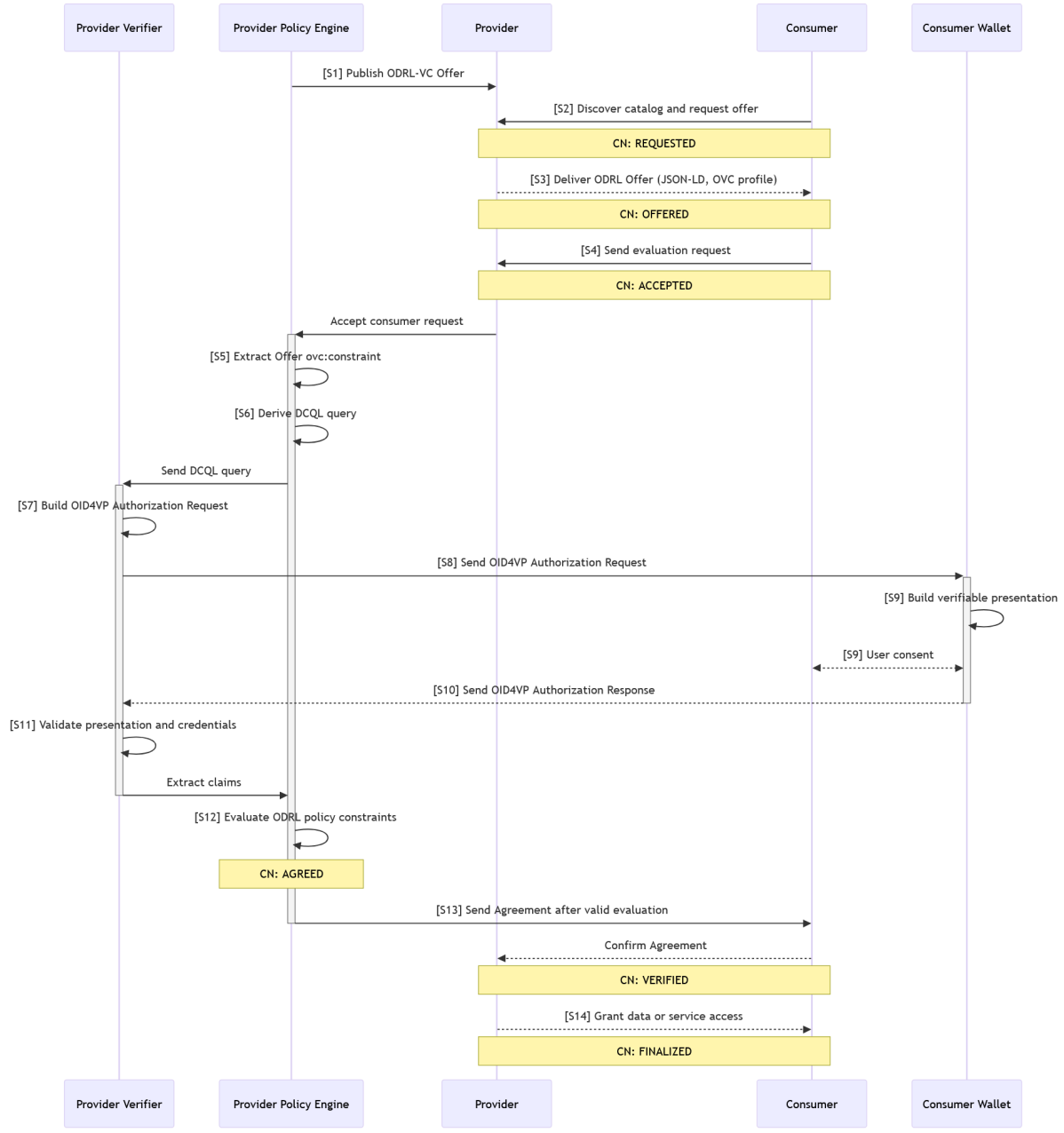

Figure 1: Sequence diagram: Provider–Consumer flow from discovery to data exchange

4.4. ODRL-to-DCQL Mapping Rules

To satisfy the determinism requirement (NFR4), we define the following normative mapping rules. Given a set $\mathcal{C} = \{c_1, \ldots, c_n\}$ of `ovc:constraint` elements extracted from an ODRL Offer, the Provider MUST construct a DCQL query Q as follows.

M1. Grouping. Partition $\mathcal{C}$ by `ovc:credentialSubjectType` value. Each distinct type t yields one credential query object q_t in Q.

M2. Credential type filter. For each q_t, set the `meta` field to filter on t using the format-appropriate type mechanism.

M3. Claim path construction. For each constraint c_i in the partition of t, add a claim selector

whose `path` array is derived from the JSONPath expression in `ovc:leftOperand`$_i$ by converting the dot-notation segments to an array of string keys:
(e.g., `$.credentialSubject.gx:legalAddress.gx:countryCode`
$\rightarrow$ `["credentialSubject", "gx:legalAddress", "gx:countryCode"]`).

M4. Operator and value metadata. ODRL operator and `rightOperand` semantics are *not* embedded in the DCQL query; they are retained in the normalized constraint tuple and applied during policy evaluation (**S12**). DCQL is responsible only for requesting the claims; ODRL governs the evaluation pass/fail semantics.

M5. Issuer restrictions. If the Data Space governance profile mandates credentials from specific issuers or trust registries, the Provider MAY add issuer filters to q_t as additional `meta` fields, beyond what is derivable from the ODRL constraints alone.

4.5. Example ODRL policy 1: regional legal entity restriction

This first example is aligned with the ODRL VC Profile restricting access to legal entities whose legal address country code is in an approved region list.

Listing 1: ODRL Offer with one ODRL-VC constraint

```
{
  "@context": [
    "http://www.w3.org/ns/odrl.jsonld",
    {"gx": "https://w3id.org/gaia-x/"},
    {"ovc": "https://w3id.org/gaia-x/ovc/1/"}
  ],
  "@type": "Offer",
  "uid": "https://provider.example/offer/1",
  "profile": "https://w3id.org/gaia-x/ovc/1/",
  "permission": [{
    "@type": "Permission",
    "target": "https://provider.example/dataset/1",
    "action": "odrl:use",
    "assigner": "https://provider.example/participant",
    "ovc:constraint": [{
      "ovc:leftOperand": "$.credentialSubject.gx:legalAddress.gx:countryCode",
      "operator": "odrl:isAnyOf",
      "rightOperand": ["FR", "BE", "ES"],
      "ovc:credentialSubjectType": "gx:LegalPerson"
    }]
  }]
}
```

Interpretation. The Provider asks the Consumer to prove possession of a `gx:LegalPerson` VC whose `countryCode` claim is one of the permitted values `(France, Belgium, Spain)`.

4.6. Example ODRL policy 2: membership level and compliance constraints

The second example demonstrates a conjunctive policy requiring the Consumer to satisfy two independent VC-based constraints drawn from two distinct credential types (a membership VC and a compliance VC).

Listing 2: ODRL Offer policy with multiple ODRL-VC constraints

```
{
  "@context": [
    "http://www.w3.org/ns/odrl.jsonld",
    {"ex": "https://example.org/vocab#"},
    {"ovc": "https://w3id.org/gaia-x/ovc/1/"}
  ],
  "@type": "Offer",
  "uid": "https://provider.example/offer/2",
  "profile": "https://w3id.org/gaia-x/ovc/1/",
  "permission": [{
    "@type": "Permission",
    "target": "https://provider.example/service/1",
    "action": "odrl:play",
    "assigner": "https://provider.example/participant",
    "ovc:constraint": [
      {
        "ovc:leftOperand": "$.credentialSubject.membership.level",
        "operator": "odrl:gteq",
        "rightOperand": 2,
        "ovc:credentialSubjectType": "ex:MembershipCredential"
      },
      {
        "ovc:leftOperand": "$.credentialSubject.compliance.status",
        "operator": "odrl:eq",
        "rightOperand": "active",
        "ovc:credentialSubjectType": "ex:ComplianceCredential"
      }
    ]
  }]
}
```

Interpretation. The Consumer must present a `ex:MembershipCredential` with level ≥ 2 (level refers to a numeric tier of membership, where usually higher tiers mean more benefits) and a `ex:ComplianceCredential` with active status.

4.7. Example Verifiable Credential: Membership

The following example shows a concrete `ex:MembershipCredential` that satisfies the membership constraint defined in the ODRL Offer of listing 2.

Listing 3: Verifiable Credential: `ex:MembershipCredential`

```
{
  "@context": [
    "https://www.w3.org/ns/credentials/v2",
    { "ex": "https://example.org/vocab#" }
  ],
  "id":   "https://consumer.example/credentials/membership",
  "type": ["VerifiableCredential", "ex:MembershipCredential"],
  "issuer":     "did:web:trusted-issuer.org",
  "validFrom":  "2025-10-06T15:44:02Z",
  "validUntil": "2026-10-06T15:44:02Z",
  "credentialSubject": {
    "https://schema.org/name": "Consumer Membership Credential",
    "membership": {
      "status": "active",
      "level":  3
    }
  }
}
```

Interpretation. The credential is issued by `did:web:trusted-issuer.org` and remains valid until October 2026. The Policy Engine applies the expression `$.credentialSubject.membership.level` to extract the value `3` and evaluates it against the `odrl:gteq` operator with `rightOperand` `2`: the constraint is satisfied. The `membership.status` claim is not required by the ODRL Offer in listing 2 and is therefore not requested in the derived DCQL query, in conformance with the data-minimization principle.

4.8. Illustrative DCQL request derived from Policy 2

The exact DCQL syntax depends on the chosen OpenID4VP profile and credential formats. The following snippet is illustrative and intended to show the mapping logic (credential type + requested claims) rather than a strict representation.

Listing 4: Illustrative DCQL query derived from listing 2

```
{
  "credentials": [
    {
      "id": "membership",
      "format": "dc+sd-jwt",
      "meta": {
        "vct_values": ["ex:MembershipCredential"]
      },
      "claims": [
        {"path": ["credentialSubject", "membership", "level"]}
      ]
    },
    {
      "id": "compliance",
      "format": "jwt_vc_json",
      "meta": {
        "type_values": ["VerifiableCredential", "ex:ComplianceCredential"]
      },
      "claims": [
        {"path": ["credentialSubject", "compliance", "status"]}
      ]
    }
  ]
}
```

Note. Applying rule **M1**, the two constraints partition into two credential groups, yielding a two-entry DCQL query (listing 4). Both credential queries are mandatory by default. The Provider requests both credentials in a single OpenID4VP transaction. ODRL remains the normative policy language for authorization semantics; DCQL expresses what to request from the Wallet. The Policy Engine performs the final ODRL evaluation after receiving the claims.

4.9. Failure Handling and Negotiation Outcomes

The protocol defines the following failure conditions and corresponding CNS outcomes.

- **No matching credential.** The Wallet cannot find a credential satisfying the DCQL type filter, the Consumer MUST terminate the exchange. The CNS transitions to state TERMINATED.
- **User consent denied.** The user rejects disclosure in the Wallet. The CNS MUST be terminated; re-initiation is permitted.
- **Cryptographic verification failure.** Any failure in step **S11** (nonce mismatch, invalid proof, revoked credential, expired credential, untrusted issuer) MUST cause the Provider to return a typed error code. The Provider MUST terminate the CNS and MUST NOT proceed to policy evaluation.

- **Policy evaluation failure.** If one or more `ovc:constraint` elements evaluate to `false`. The Provider MUST return a termination message that identifies the failing `ovc:leftOperand` paths so the Consumer can determine which claims are insufficient.

5. Discussion

5.1. Semantic Coherence Between Policy and Proof Request

A primary motivation for the proposed composition of standards is the elimination of semantic drift between the authorization policy and the identity verification configuration. In typical deployments, a Verifier's credential request is configured independently of the access policy, creating a maintenance burden and the risk of requesting credentials that do not match the actual policy conditions. By deriving the DCQL query deterministically from the ODRL-VC constraints (rules **M1–M5**), the proposed protocol ensures that the proof request is always consistent with the Offer policy in force. A policy update automatically propagates to the DCQL query at the next negotiation; no separate verifier reconfiguration is required.

The approach also supports the implementation of Data Space governance: a provider can publish offers with machine-testable eligibility conditions, and consumers can prove eligibility during negotiation rather than after transfer setup.

5.2. Privacy and Data Minimization

Mapping rule **M3** ensures that the DCQL query requests only the claim paths explicitly referenced by `ovc:leftOperand` expressions. When the Consumer Wallet supports selective disclosure (e.g., via SD-JWT-VC [16]), only those claims are included in the `vp_token`; the remainder of the credential payload is withheld. This is consistent with the data-minimization obligations under the GDPR [19] applied to identity data sharing: a Consumer does not need to disclose their full organizational profile to satisfy a single jurisdictional constraint, making this protocol particularly suitable for contexts involving personal data or stringent privacy requirements.

5.3. Trust Framework Considerations

The ODRL-VC Profile identifies claim paths and credential types but does not specify which issuers are trusted or how credential status is managed. Therefore, the following questions arise:

- Which issuers are trusted?
- Which schemas/context versions are accepted?
- How are revocation/status lists checked?
- Are credentials required to be recent (issuance time constraints)?

In practice, these elements must be provided by the Data Space governance layer. Concrete questions include: which issuers are listed in the applicable trust registry; which credential schemas and JSON-LD contexts are accepted; whether revocation is checked via Status List 2021 [17] or another mechanism; and whether a maximum credential age applies. We note that some of these constraints can be expressed as additional ODRL constraints (e.g., an obligation to present a credential issued within the last 30 days), while others are more naturally encoded as verifier-side metadata (mapping rule **M5**). Data Spaces SHOULD document this split explicitly in their governance profiles.

5.4. Interoperability Edge Cases

Several issues need careful profiling:

- **JSONPath consistency**: Different JSONPath engines may differ at the edges. A deeper integration between the ODRL-VC Profile and DCQL could further reduce this risk by allowing claim paths to be expressed directly as DCQL path arrays, eliminating the JSONPath implementation, ODRL to DCQL translation step and the associated engine inconsistencies entirely.
- **Credential format diversity**: OpenID4VP supports multiple credential formats; claim extraction paths may vary (e.g., `SD-JWT-VC` vs `JWT VC JSON` vs `mdoc`).
- **Multi-credential policies**: Some policies require conjunction across credentials; others allow alternatives. This should be explicit in the ODRL policy or a companion mapping profile.

5.5. Security considerations

Implementations MUST strictly conform to the OpenID4VP specification [9], in particular the nonce and audience binding requirements, holder proof verification, and `client_id` validation, which together prevent replay, cross-negotiation substitution, and impersonation attacks. All cryptographic checks on Verifiable Credentials MUST be performed in full before policy evaluation is attempted; partial verification MUST NOT result in an access grant. A tamper-evident audit trace MUST be maintained for every negotiation, recording the policy version, the derived DCQL query, the received `vp_token` digest, and the per-constraint evaluation outcome, keyed by CNS transaction identifier. The only EXCEPTION concerns credential claims that constitute personal data under the GDPR [19]: such claims MUST NOT be retained beyond the immediate evaluation context, and their processing MUST comply with applicable data protection obligations, including the purpose limitation and storage limitation principles of Article 5(1)(b) and (e).

6. Conclusions

This paper proposed a concrete, standards-based protocol for credential-governed contract negotiation in Data Spaces. The protocol consists of five complementary specifications — W3C Open Digital Rights Language (ODRL) [3, 4], W3C Verifiable Credentials [7], the ODRL-VC Profile [8], OpenID4VP [9], and the Dataspace Protocol [2] — into a coherent end-to-end flow in which the Provider publishes an ODRL Offer with machine-verifiable VC-based constraints, and the Consumer proves compliance by presenting credentials from a Wallet before a binding agreement is issued.

The key contributions are: (i) a formal 14-step interaction sequence that integrates VC presentation into the DSP contract negotiation state machine; (ii) five normative ODRL-to-DCQL mapping rules that deterministically derive a minimal credential request from an ODRL-VC Offer, satisfying both the data-minimization principle and the determinism requirement; and (iii) policy examples illustrating single-credential and multi-credential scenarios.

Several directions remain open for future work. First, the DCQL mapping rules should be profiled more strictly for different credential formats, and the ODRL-VC Profile itself could be extended to align more closely with the DCQL query model, reducing the translation layer between policy constraints and credential requests. Second, the interaction between trust framework metadata and ODRL constraints could be formalized in a companion governance profile, so that implementers have unambiguous guidance on the boundary between policy-level and verifier-level configuration. Third, extending the protocol to support relationship-based access control scenarios, where eligibility depends on organizational roles or party relationships rather than attribute values alone, would broaden its applicability, complementing profile proposals that introduce such constructs into the ODRL vocabulary for Data Spaces [20]. Fourth, the automatic generation of ODRL-VC Offer policies from natural language instructions using ontology-guided large language models [21] could lower the authoring barrier significantly,

bringing the full pipeline from human-readable access requirement to cryptographically verifiable negotiation closer to practical deployment. Finally, a reference implementation and interoperability test suite would provide empirical validation of the protocol's determinism and correctness claims across heterogeneous Data Space deployments.

Acknowledgments

The authors would like to thank Giuditta del Buono (Gaia-X AISBL, Brussels, Belgium) for her valuable comments and suggestions, which greatly contributed to the refinement of this paper.

Declaration on Generative AI

During the preparation of this work, the authors used a generative AI assistant to support drafting, restructuring, paraphrasing, and improving the clarity and readability of the text. After using this tool, the authors thoroughly reviewed and edited all content as needed and take full responsibility for the publication's content.

References

[1] Data Exchange Working Group, Data Usage Agreement, Technical Report, Gaia-X Association AISBL, 2025. URL: https://docs.gaia-x.eu/technical-committee/data-exchange/25.07/data-usage-agreement/.

[2] Eclipse Dataspace Working Group, Dataspace Protocol Specification, Technical Report, Eclipse Dataspace Working Group, 2025. URL: https://eclipse-dataspace-protocol-base.github.io/DataspaceProtocol/2025-1-err1/.

[3] R. Iannella, S. Villata, ODRL Information Model 2.2, W3C Recommendation, World Wide Web Consortium, 2018. URL: https://www.w3.org/TR/odrl-model/.

[4] R. Iannella, M. Steidl, ODRL Vocabulary & Expression 2.2, W3C Recommendation, World Wide Web Consortium, 2018. URL: https://www.w3.org/TR/odrl-vocab/.

[5] Pierre Gronlier, Gaia-X Trust Framework, Technical Report, Gaia-X Association AISBL, 2023. URL: https://docs.gaia-x.eu/#/framework.

[6] D. S. S. Centre, DSSC Blueprint V3.0: Building on Top of Foundational Standards – Section 3.3: Open Digital Rights Language (ODRL), Technical Report, Data Spaces Support Centre, 2025. URL: https://blueprint.dssc.eu/?pane=technical&technical=building-on-top-of-foundational-standards#BuildingonTopofFoundationalStandards-3.3OpenDigitalRightsLanguage(ODRL).

[7] M. Sporny, D. Longley, D. Chadwick, O. Terbu, Verifiable Credentials Data Model v2.0, W3C Candidate Recommendation, World Wide Web Consortium, 2024. URL: https://www.w3.org/TR/vc-data-model-2.0/.

[8] Yassir SELLAMI, ODRL Verifiable Credential Profile (ODRL-VC Profile), Technical Report, Gaia-X AISBL, 2024. URL: https://w3id.org/gaia-x/ovc/1/.

[9] O. Terbu, T. Lodderstedt, K. Yasuda, T. Looker, OpenID for Verifiable Presentations, Specification 1.0, OpenID Foundation, 2024. URL: https://openid.net/specs/openid-4-verifiable-presentations-1_0.html.

[10] T. Lodderstedt, B. Campbell, N. Sakimura, Digital Credentials Query Language (DCQL), Draft Specification Draft 03, OpenID Foundation, 2024. URL: https://openid.net/specs/openid-dcql-1_0.html.

[11] A. Cimmino, J. Cano-Benito, R. García-Castro, Open Digital Rights Enforcement framework (ODRE), Computers & Security 150 (2025). doi:10.1016/j.cose.2024.104282.

[12] S. Wiedemann, ODRL-PAP, Technical Specification, FIWARE, 2026. URL: https://github.com/wistefan/odrl-pap/blob/main/README.md.

[13] W. Slabbinck, J. R. Meléndez, B. Esteves, R. Verborgh, P. Colpaert, May the FORCE be with you? A framework for ODRL rule compliance through evaluation, in: 2nd NeXt-generation Data Governance workshop (NXDG 2025), co-located with the 21th SEMANTiCS conference, 2025.
[14] W. Slabbinck, J. Rojas Meléndez, B. Esteves, P. Colpaert, R. Verborgh, Interoperable Interpretation and Evaluation of ODRL Policies, in: E. Curry, M. Acosta, M. Poveda-Villalón, M. van Erp, A. Ojo, K. Hose, C. Shimizu, P. Lisena (Eds.), The Semantic Web, Springer Nature Switzerland, Cham, 2025, pp. 192–209. doi:10.1007/978-3-031-94578-6_11.
[15] B. Esteves, W. Slabbinck, Y. Sellami, A. Cimmino, V. Rodriguez-Doncel, R. Verborgh, Capturing Requests and Context for ODRL-based Access and Usage Control, in: 16th Workshop on Ontology Design and Patterns (WOP 2025), co-located with the 24th International Semantic Web Conference (ISWC 2025), 2025. URL: https://ceur-ws.org/Vol-4093/paper5.pdf.
[16] D. Fett, B. Campbell, J. Bradley, T. Lodderstedt, M. Jones, D. Waite, SD-JWT-based Verifiable Credentials (SD-JWT VC), IETF Internet Draft draft-ietf-oauth-sd-jwt-vc-05, Internet Engineering Task Force, 2024. URL: https://drafts.oauth.net/oauth-sd-jwt-vc/draft-ietf-oauth-sd-jwt-vc.html.
[17] M. Sporny, D. Longley, Bitstring Status List v1.0, W3C Candidate Recommendation, World Wide Web Consortium, 2024. URL: https://www.w3.org/TR/vc-bitstring-status-list/.
[18] S. Gössner, G. Normington, C. Bormann, JSONPath: Query Expressions for JSON, RFC 9535, Internet Engineering Task Force, 2024. URL: https://www.rfc-editor.org/rfc/rfc9535.
[19] European Parliament, Council of the European Union, General Data Protection Regulation (GDPR), Regulation (EU) 2016/679, Official Journal of the European Union, 2016. URL: https://eur-lex.europa.eu/eli/reg/2016/679.
[20] I. Plaza-Ortiz, A. Munoz-Arcentales, J. Salvachúa, C. Aparicio, G. Huecas, E. Barra, Authentication and authorization in data spaces: A relationship-based access control approach for policy specification based on odrl, 2025. URL: https://arxiv.org/abs/2505.24742. arXiv:2505.24742.
[21] D. M. Mustafa, A. Nadgeri, D. Collarana, B. T. Arnold, C. Quix, C. Lange, S. Decker, From instructions to ODRL usage policies: An ontology guided approach, 2025. URL: https://arxiv.org/abs/2506.03301. arXiv:2506.03301.