Proofs over Statements in Verifiable Evidence Graphs for the European Business Wallet

Dr. Carsten Stöcker
Spherity GmbH, Düsseldorf, Germany
carsten.stoecker@spherity.com

Prepared: 9 November 2026



Except where otherwise noted, this work is licensed under the Creative Commons Attribution 4.0 International License (CC BY 4.0). © 2026 Carsten Stöcker, Spherity GmbH.

Abstract

Industrial, agentic and physical AI systems use evidence issued by other organisations. A verifiable evidence graph preserves the source, meaning and version of each assertion when evidence is combined. The European Business Wallet proposal provides for linked attestations and mandates [1]. We examine what a relying party must retain to verify assertions after a merge.

Finite named facts with set-union composition admit an RDF encoding. Graph proofs support verification from recoverable source datasets and proof information; envelopes require their protected signing inputs, relevant disclosures and a governed graph mapping. Under some disclosure policies, withholding a type changes the expanded graph. For synthetic IRI-only credentials of 17–574 quads, the cost model estimates crossovers of 1.32–4.84 additional verifications when envelope checks refresh that mapping. Checking only the retained envelope can cost less under a trusted mapping. A separate presentation test favours the tested SD-JWT implementation under its different proof procedure. An attack model bounds decision assurance by feasible substitution costs; an authority check supplies an upper bound when changing it alone enables an accepted substitute that changes the outcome. For retained, multi-issuer evidence requiring repeated graph verification, we recommend a common W3C VCDM 2.0 JSON-LD profile secured by RDF-based Data Integrity. Additional envelopes should serve a defined presentation, disclosure or interoperability purpose. Qualified use requires an eligible profile and qualified issuance. The examined EUDI Wallet issuance baseline references SD-JWT VC and mdoc profiles [2, 3]; neither format is restricted to one-time use. Worked designs for battery-passport analytics, customs release and an aerospace safety supervisor motivate an additional graph-proof issuance option. These conditional results do not establish universal performance or security superiority.

Keywords: verifiable credentials; linked data; RDF canonicalisation; JSON-LD; SD-JWT; eIDAS; European Business Wallet; qualified electronic attestation of attributes; evidence graphs; AI safety; runtime assurance; digital product passport; post-quantum cryptography; hybrid signatures

Version 1.3, 9 November 2026. Final editorial review and PQC extension. Licence: CC BY 4.0.

Contents

1 Introduction **4**

1.1 The problem 4

1.2 The regulatory setting 4

1.3 Research questions and contributions 4

1.4 Summary of results 5

2 Background **6**

2.1 Linked data in one page 6

2.2 W3C Verifiable Credentials 7

2.3 IETF SD-JWT VC and ISO mdoc 8

2.4 The eIDAS trust layer 8

2.5 The EUDI Wallet baseline and the EBW proposal 8

2.6 The global landscape 9

3 Industry requirements **11**

3.1 Yardstick 11

3.2 Domains 11

3.3 Cross-domain properties 13

4 Formal model **13**

4.1 Preliminaries 13

4.2 Objects 14

4.3 Proof domains 15

4.4 Requirements as conditions 16

5 Results **17**

5.1 An RDF encoding of the requirements 17

5.2 What the proof covers 18

5.3 Selective disclosure 21

5.4 Complexity as a security property 22

5.5 Cost at scale 23

5.6 Assurance propagation 29

6 Use cases **33**

6.1 UC1: supply-chain analytics on digital product passports 33

6.2 UC2: e-commerce customs release 36

6.3 UC3: an AI safety supervisor in aerospace 39

7 Related work **42**

8 Discussion **42**

8.1 Expandable, source-backed and graph-native 42

8.2 Why qualification matters, and what should be qualified 43

8.3 What does not change for qualified trust service providers 44

8.4 Presentation and reuse 44

8.5 Global interoperability 45

8.6 Hybrid signatures and post-quantum migration 45

9 Regulatory and standards path **46**

10 Limitations **46**

11 Conclusion **47**

Declarations **48**

Appendix A. Proof remarks **49**

Appendix B. Benchmark **49**

Appendix C. Complete signed credentials and verification **51**

References **56**

1 Introduction

1.1 The problem

Consider three automated decisions. A purchasing agent places an order on behalf of a company, relying on a supplier's certification and on the mandate under which it acts. An analytics service reports a battery's carbon-footprint class, relying on a manufacturer's declaration and a verification statement. A safety supervisor permits an action proposed by an AI agent, relying on regulatory, standard and internal requirements. Each decision depends on evidence from several organisations.

A company can set the rules for evidence produced within its own systems. Across companies, the relying party must establish where the evidence came from, what it asserts and which authority it carries. Products can outlive their suppliers, and automated consumers need these checks to be explicit. We call the resulting structure a *verifiable evidence graph*. Its assertions link to their sources, interpretation and versions. For each operation, the relying party can check the actor's authority and current permission.

1.2 The regulatory setting

The European Business Wallet (EBW) proposal [1] provides for linked attestations forming part of a chain (Art. 5(1)(g)), authorised representatives acting under mandates (Art. 3(18)–(19)), and roles, mandates and delegation for access control (Annex, section 12). Recital 4 envisages reuse of European Digital Identity (EUDI) specifications subject to EBW-specific rules. Proposed Art. 8(3) refers to those specifications for owner identification. For qualified electronic attestations of attributes (QEAA) and attestations issued by public sector bodies (PuB-EAA) within the current EUDI Wallet issuance rules, the implementing regulations reference the SD-JWT VC and mdoc profiles. The W3C Verifiable Credentials Data Model (VCDM) profile in ETSI TS 119 472-1 clause 7 is outside the reference. Section 2.5 traces the operative provisions [2–4].

A proposed way to carry linked data within an existing credential format is a *nested hybrid*. A JSON-LD document, which denotes a graph, is placed inside an SD-JWT. The payload remains expandable to RDF. The question is what a relying party must retain and check to verify a statement taken from that payload after a merge, and what that costs. We analyse the construction without assuming that an arbitrary nested credential qualifies under the issuance rules.

1.3 Research questions and contributions

- **RQ1.** What do industrial application domains require from attestations exchanged through the EBW?
- **RQ2.** What does a verifier need to retain when the proof covers the graph or when it covers a nested payload?
- **RQ3.** How does verification cost vary with reuse in each architecture?
- **RQ4.** What assurance level can an automated decision reach when its evidence graph mixes qualified, non-qualified and unsigned facts?

Contributions

1. A requirements synthesis across eight domains (business-to-business exchange, digital twins, machine-to-machine exchange, digital product passports, agent-to-agent interaction, safe AI, industrial AI, physical AI), plus cross-border trade (Section 3).
2. A formal model of evidence graphs with pluggable trust anchors, assurance levels and admissible formats, and the notion of a securing mechanism's *proof domain* (Section 4).
3. Results on proof recovery, context resolution, selective disclosure, reuse cost and decision assurance under a stated attack model (Section 5).
4. A benchmark of full-disclosure processing with an in-memory quad store, comparison timings for published libraries, selective-disclosure timings and a cost model (Section 5.5, Appendix B).
5. Three worked designs with cost estimates (Section 6) and a standards proposal (Section 9).

1.4 Summary of results

Finite, globally named facts with set-union composition admit an RDF encoding (T1). A flat profile without shared identifiers and predicate mappings cannot provide those properties directly. RDF is the standards choice here; T1 establishes representability, not a unique solution or a performance advantage. Under the graph-proof profile, a verifier can recover the original proof input from a lossless store of credential quads and proof information. An ordinary envelope verifier must recover the signed input and any openings relevant to the statement, for example from a retained artefact. External context changes can alter the expanded graph under a still-valid envelope signature. Under fixed processing rules, a graph proof detects changes to its authenticated quads (L1, L2). A relying party can also use a trusted delegated result or issuer authentication of the graph form (L3).

For the IRI-only fixtures, the cost model favours the graph path from the second to the fifth additional check onwards, depending on size, when the envelope path refreshes the mapping each time (T2). The source-only path checks the retained envelope and assumes the earlier mapping can be trusted. It can cost less, and it answers a different verification question. The presentation test compares two published implementations with different proof structures and verification procedures (Sections 5.3 and 5.5).

In the attack model, a feasible change to one slot caps assurance if it changes the outcome (T3). An authority check gives such a bound when the changed authority alone permits an accepted substitute that changes the outcome. Qualification can address these attacks if the policy and attack costs justify the target. Joint attacks can impose further requirements. A relying party that requires one graph-proof pipeline also needs an eligible graph-proof issuance profile for any required qualified attestations (C1). Across borders, technical validation and legal recognition are separate (C2). Figure 1 shows the results and the premises they use.

Cryptographic migration. Post-quantum cryptography (PQC) adds another design dimension. A hybrid signature combines cryptographic algorithms; the nested hybrid studied here combines credential representations. These are different constructions. W3C Data Integrity already supports multiple signatures through proof sets and chains [5, 6]. Using them for a traditional/PQC combination still requires a validated hybrid design, mandatory-component policy and downgrade protection [7]. Section 8.6 explains the verification work; the reported benchmarks do not measure it.

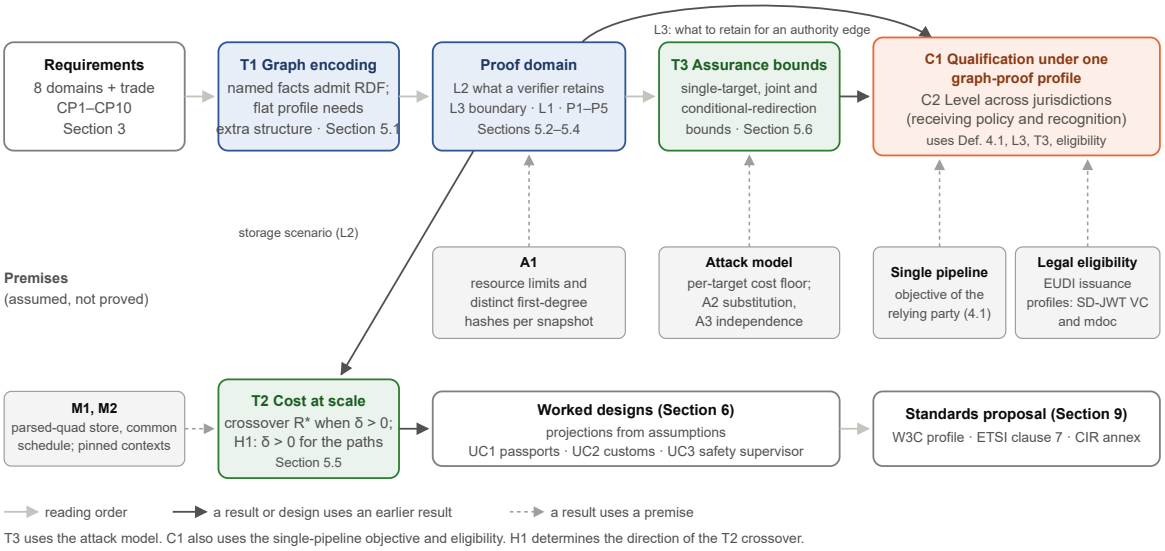


Figure 1: Structure of the argument. Rounded boxes are results. Grey boxes identify modelling conditions and premises. Arrows show how the results and worked designs are related.

2 Background

This section is written for readers from industry, law and policy who are not specialists in linked data or credential formats.

2.1 Linked data in one page

A *statement* (or triple) has a subject, a predicate and an object. For example, battery 4711 has a recycled-cobalt share of 16 %. In RDF, the W3C's data model for linked data, a predicate is identified by an IRI, such as a web address, and a subject is identified by an IRI or a blank node [8]. A blank node represents a resource without giving it a global name. If two organisations use the same identifier for a battery and share the vocabulary for its properties, they can state facts about the battery independently and combine them. A set of triples forms a graph. Under RDF simple semantics, adding statements preserves earlier entailments [9]. An application's decision can still change when it sees new evidence or rejects revoked evidence. A shared vocabulary defines terms used in statements, as in GS1 EPCIS or the Asset Administration Shell [10, 11].

JSON-LD is a JSON syntax for linked data [12]. A *context* can map short JSON keys to IRIs. Expansion and conversion to RDF then give the statements used by a graph application.

A running example. A recycler and a battery manufacturer identify a battery as https://id.example/battery/4711. In this example the context maps id to @id and recycledCobaltShare to https://vocab.example/bp#recycledCobaltShare. The following unsigned JSON-LD claim fragment illustrates the recycler's assertion:

```
{
  "@context": {
    "id": "@id",
    "recycledCobaltShare": "https://vocab.example/bp#recycledCobaltShare"
  },
  "id": "https://id.example/battery/4711",
  "recycledCobaltShare": 16
}
```

The corresponding statement has the battery identifier as subject, the property IRI as predicate and the integer 16 as object. A manufacturer's record using the same identifier adds statements about the same subject. When the claim is secured, an issuer proof lets a verifier check that the recycler signed the assertion and that it has not changed. A separate authority check establishes whether the policy accepts that recycler for the assertion, for example through a register extract and a credential covering the scope of its work. The integer alone does not encode a unit; the illustrative vocabulary must define whether this share is expressed in percentage points. This listing and the two witnesses in Section 5.2 are minimal semantic fragments, not complete signed credentials. A complete W3C VCDM 2.0 credential requires the prescribed context, credential type, issuer and credential subject, together with a securing mechanism for verification; those surrounding fields and proofs are omitted here. Appendix C supplies complete signed plain and nested credentials, their public verification material and executed checks.

Public web pages illustrate the use of shared vocabularies and JSON-LD. Google, Microsoft and Yahoo launched schema.org in 2011. Yandex added support that year [13, 14]. Google recommends JSON-LD for supported structured data [15]. The HTTP Archive Web Almanac 2025 found structured data on 50 % of sampled home pages and JSON-LD on 43 % of them [16]. Web Data Commons reports about 2.39 billion parsed URLs, 16.5 million pay-level domains with extracted data and 74 billion RDF quads for the October 2024 Common Crawl [17].

Google reported over 500 billion facts about five billion entities in its Knowledge Graph in 2020 [18]. It describes the graph as one source for AI Mode and says that inclusion in its AI features requires no special markup [19, 20]. Wikidata provides a public knowledge graph and a SPARQL query service. Its statistics page reported about 124 million items when checked in November 2026 [21]. These examples describe distinct systems. The crawl figures measure the markup found on pages. Figure 2 uses ordinary self-asserted markup without an attached issuer proof, version or status record as its starting point.

2.2 W3C Verifiable Credentials

A W3C Verifiable Credential expresses claims about one or more subjects and provides a way to verify the issuer's assertion. Version 2.0 of the data model became a W3C Recommendation on 15 May 2025 [22]. Two families of securing mechanisms are relevant here. With a graph-domain Data Integrity cryptosuite such as eddsa-rdfc-2022, the issuer converts the credential and relevant proof options to RDF, canonicalises the input, and signs its hashed representation [5, 6, 23]. W3C VC JOSE/COSE specifies securing the JSON-LD document with JWS, SD-JWT or COSE [24]. A decentralised identifier (DID) can resolve to a document listing verification methods. Issuer authority and credential status require separate checks [22, 25]. Data Integrity also has suites for selective disclosure. ecdsa-sd-2023 is a Recommendation, and the BBS cryptosuites were a Candidate Recommendation Draft on 10 September 2026 [26, 27].

Data Integrity proof sets group independent proofs; proof chains record dependencies through previousProof. These support multiple signatures within one credential without adding an outer

credential envelope [5, 6]. Multiple signatures do not by themselves provide threshold signatures, post-quantum security or legal qualification. Their required combination must be specified by the application profile.

2.3 IETF SD-JWT VC and ISO mdoc

SD-JWT (RFC 9901, November 2025) uses a signed JWT to authenticate cleartext claims and commitments to selected JSON members or array elements. The holder reveals a committed member or element by presenting its salted disclosure. Where required, a key-binding JWT proves possession of the holder's key for the presentation [28]. IETF SD-JWT VC profiles SD-JWT for credentials and uses a vct credential type. The Internet-Draft was at version 26 on 6 November 2026 [29]. It uses dc+sd-jwt, while W3C VC JOSE/COSE uses vc+sd-jwt for its SD-JWT securing mechanism [24]. The IETF profile does not require JSON-LD processing. ISO/IEC 18013-5 mdoc uses CBOR, a docType and an issuer-signed object with digests of individually salted data elements. Presentations can include device authentication [30, 31]. Both formats support selective disclosure of identity attributes.

2.4 The eIDAS trust layer

Regulation (EU) No 910/2014, as amended by Regulation (EU) 2024/1183 [32], provides for supervision and conformity assessment of qualified trust service providers (QTSPs). National trusted lists identify qualified services. Article 45b(1) prevents denial of legal effect or admissibility in legal proceedings solely because an electronic attestation of attributes (EAA) is electronic or does not meet the requirements for a qualified EAA. A qualified EAA (QEAA) is issued by a QTSP and must meet Annex V. An EAA issued by or on behalf of a public sector body responsible for an authentic source (PuB-EAA) must meet Article 45f and Annex VII. Both have the same legal effect as lawfully issued paper attestations under Article 45b(2). Recognition across Member States follows Article 24a(9) for a QEAA and Article 45b(3) for a PuB-EAA. Section 4.2 uses level Q as shorthand for these two categories.

Article 45e requires Member States to enable QTSPs to verify at least the attributes in Annex VI electronically at the user's request, where those attributes rely on public-sector authentic sources, in accordance with Union or national law. Annex VI includes company data and powers and mandates to represent natural or legal persons. Conformity-assessment and accreditation certificates are not expressly listed. A qualified electronic seal carries a presumption of data integrity and correct origin under Article 35(2) and can secure an attestation. Qualified electronic ledgers have a separate presumption for the integrity and chronological ordering of their records under Articles 45k and 45l. Article 14 provides for the recognition of trust services from third countries or international organisations through implementing acts or agreements [32].

2.5 The EUDI Wallet baseline and the EBW proposal

For in-scope QEAA and PuB-EAA issuance, Annex II of Implementing Regulation (EU) 2025/1569, as amended by 2026/1735, refers to the standards in Annex II of Implementing Regulation (EU) 2024/2979. Implementing Regulation (EU) 2026/1731 replaces that annex with a reference to clauses 2–6 of ETSI TS 119 472-1 V1.2.1, with adaptations. Clauses 5 and 6 specify the SD-JWT VC and ISO mdoc formats and distinguish general EAAs from QEAA and PuB-EAA, including which entities their attributes may concern. A relationship can be expressed as an attribute. An arbitrary JSON-LD payload inside an SD-JWT VC still needs to meet the applicable profile to qualify. Clause 7, outside the current legal reference, already

addresses W3C VCDM, including embedded proofs, selective disclosure and bindings to qualified signatures or seals [2–4, 33].

Issuance and presentation are separate. For presentation, 2026/1731 Annex XII references clauses 4.1, 4.2, 5 and 6 of ETSI TS 119 472-2 V1.2.1, with adaptations. The specification also contains W3C JOSE presentation and request annexes outside that reference. Its OpenID4VC profile draws on HAIP 1.0, which profiles SD-JWT VC and mdoc and references SD-JWT VC draft 13 [2, 31, 34].

The EBW remains a proposal as of 8 November 2026. Recital 4 anticipates reuse of the EUDI specifications, subject to EBW-specific rules. Article 8(3) of the Commission proposal refers to the Annex II standards for EBW owner identification data. The ETSI work item for TR 119 060, a gap analysis for the EBW, lists 28 November 2026 as the planned publication date [1, 35, 36].

2.6 The global landscape

Table 1 compares selected published specifications, regulatory requirements and trade pilots as of 8 November 2026. A pilot shows what was tested. A specification shows what its authors propose or require within its scope.

Table 1: Selected uses of identifiers and credentials in trade and identity (status 8 November 2026).

Jurisdiction	**Work**	**Data model and proof**	**Status**
EU, EUDI Wallet issuance baseline	SD-JWT VC and ISO mdoc profiles of TS 119 472-1 [2, 3, 33]	SD-JWT VC or mdoc, with envelope proofs	binding for attestations in scope
EU, product data	DPP registry semantic guide on RDF, JSON-LD, SKOS, SHACL and OWL. The registration API uses JSON [37, 38]	RDF/JSON-LD in the semantic guide	registry launched on 20 July 2026
UN/CEFACT	UNECE Recommendation 49 (adopted July 2025) and the separate UN Transparency Protocol 0.7.0 specification [39, 40]	UNTP specifies VCDM 2.0 in compacted JSON-LD, with an enveloping JOSE proof	recommendation and pre-production pilot specification
US, DHS S&T	VC/DID implementation profile (work in progress, 2022) and reported plugfest [41, 42]	compacted JSON-LD and linked-data proofs in the draft profile	draft profile and experimentation

Continued on next page

Jurisdiction	Work	Data model and proof	Status
US, CBP	August 2023 mock test with steel and oil data (over 1,000 VCs). The ACE 2.0 issue paper identifies VCs and DIDs as technologies CBP is pursuing. A voluntary crude-oil pipeline test was announced in 2026 [43–45]	VCs and DIDs. These sources do not specify a common credential encoding	pilots and stated direction
US, NIST	SP 800-63C-4 gives examples of attribute bundles. Draft IR 8480 distinguishes W3C JSON-LD VCs from SD-JWT VC [46, 47]	examples include W3C VCs, SD-JWT and mdoc	guidance and draft report
US, pharmaceutical supply chain	The PDG blueprint describes an optional credential-based route under the Open Credentialing Initiative (OCI), alongside alternatives [48, 49]	OCI specifies VCDM 1.0 with embedded linked-data proofs and a JWT presentation for product-identifier verification	published industry specification
US, state identity	California mDL hackathons [50]	ISO mdoc and W3C VC were among the approaches explored	pilot programme events
China, reported initiative	A 2023 report describes RealDID and its use of blockchain for identity management [51]	reported DLT-based design	secondary reporting. Deployment and format conformance are unverified
China and international standards	CAICT and Ant Group are listed in the W3C DID Working Group [52]. ITU-T publishes recommendations on decentralised identity, including Y.3087, supported by Chinese organisations [53]	participation and published recommendations	standards work
China, cross-border	China–Singapore electronic bill of lading pilot and extension of the trade framework to the China–Tanzania corridor [54, 55]	reports describe digital trade documents and provenance. The credential encoding used in each corridor is unspecified	pilot and reported extension

The UNTP, DHS and OCI specifications show how W3C credentials can serve different trade and supply-chain designs. CBP states that it is pursuing VCs and DIDs as interoperability technologies. The cited pilots establish where the approaches have been tested. The standards entries describe participation and published specifications. Neither source type establishes a common credential encoding or the extent of national adoption.

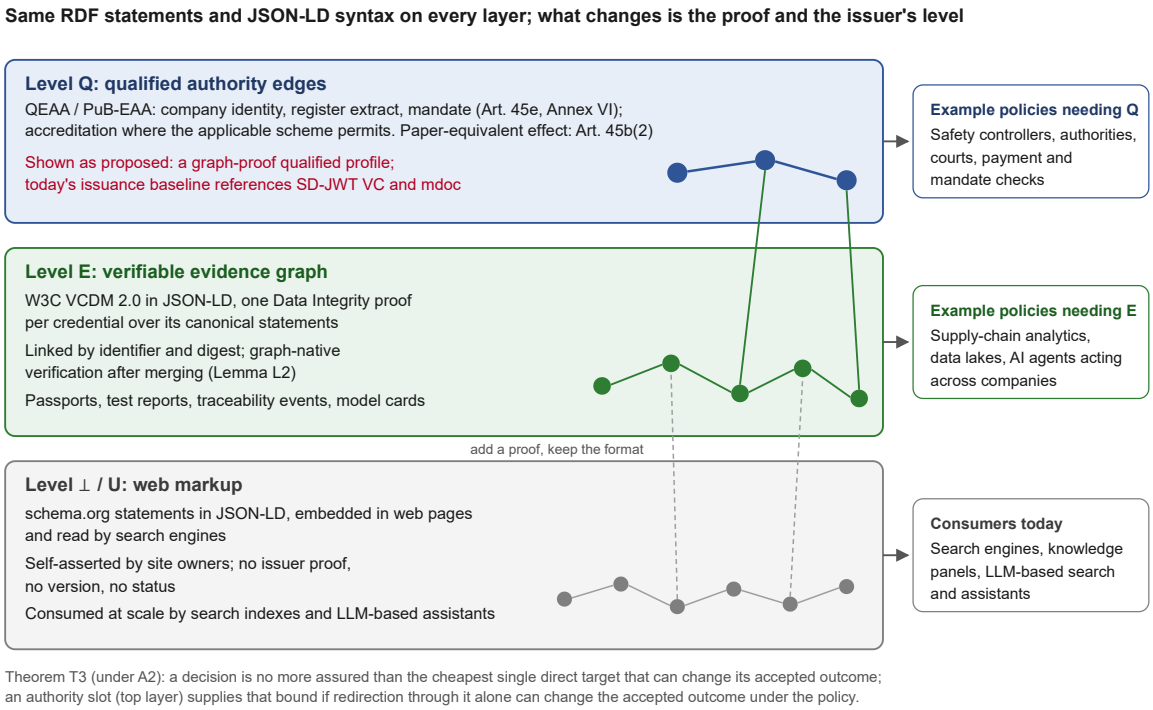


Figure 2: Web markup (schema.org in JSON-LD), a verifiable evidence graph and qualified authority edges share one data model. The qualified layer shows the graph-proof profile proposed in Section 9.

3 Industry requirements

3.1 Yardstick

The analysis uses a common design goal.

Access-controlled, verifiable evidence graphs for physical products, virtual products and physical AI. Every fact links to its source, meaning and version; every operation links to an actor's authority and current permission. Built on open international standards and open-world semantics, as shared infrastructure across companies and jurisdictions.

In this paper, verifiable evidence can be checked for attribution, integrity, version and status. A valid proof establishes what the issuer authenticated. The relying party still needs to decide whether the assertion is accurate and relevant. Its policy can use internal assertions with provenance for work within one trust domain, require issuer authentication across domains, and require qualified attestations for selected slots. Section 4.2 defines the assurance levels used in the model. Selective disclosure is a separate requirement wherever the verifier should see only part of the evidence. Section 5.6 asks which evidence checks matter for a decision, and L3 examines how those checks can be combined across formats.

3.2 Domains

We selected examples from public legal texts, standards and sector specifications and grouped them by their use in the eight domains. Cross-border trade is considered separately. The table distinguishes enacted requirements, proposed or specified capabilities, and design assumptions. Section 4.1 of the supplementary 01-requirements.md records the domain-by-property mapping.

It is a qualitative synthesis. The analysis below uses the identified properties as design conditions for evidence shared across organisations.

Table 2: Selected requirements by domain.

Domain	**Requirement**	**Source**
B2B	linked attestations, mandates and access control with delegation	Legislative proposal [1], Arts. 3(18)–(19), 5(1)(g) and Annex §12
B2B	legal effect of QEAA and verification against authentic sources for listed attributes under specified conditions	Law [32], Arts. 45b and 45e, Annex VI
Digital twins	linked submodels and concept descriptions, with an RDF mapping	Specification [10]
M2M	shared field semantics and bounded verification cost per message	Design assumptions. [56] documents a data-space wallet profile
DPP	decentralised passport data and a central registry. Covered batteries require passports from 18 February 2027, with information specified in Annex XIII. The Commission documents the registry semantics separately	Law [57, 58] and registry documentation [37]
DPP	passports, conformity credentials and traceability events as VCs in UNTP, and JSON-LD support for EPCIS events	Specifications [11, 39]
Agent-to-agent	trace an action to an accountable person and check the agent's authority outside the model	Design assumptions. [1] addresses representation and mandates in the wallet proposal
Safe AI	dataset governance and record-keeping for high-risk systems where applicable. A monitor bounds the behaviour of a complex component in the worked design	Law [59], Arts. 10 and 12. Design pattern [60]
Industrial AI	vulnerability handling for in-scope products with digital elements during the support period. Treat models and datasets as versioned artefacts with provenance in the worked design	Law [61], Art. 13(8). Provenance is a design assumption
Physical AI	specified self-evolving machine-learning systems for safety functions, and reconstructable action records in the worked design	Law [62, 63], Annex I Part A, points 5–6 and Art. 54. Action records are a design assumption motivated by [59, 64]
Cross-border	check evidence without destructive conversion, using identifiers the relying party can resolve	Design assumption informed by Table 1 and [25]

Continued on next page

Domain	Requirement	Source
Cross-border e-commerce	verified legal-entity identity of every submitting party, linked product, order, packing and transport credentials per shipment, and personal data held outside the shared graph; certificates of conformity for children's products from accepted laboratories	Public pilot demonstration [65]; law [66, 67]

3.3 Cross-domain properties

Property	Formal name or related result
CP1 relational, multi-subject facts	R-G
CP2 shared identity across issuers under a common identifier convention	R-I
CP3 order-independent, monotone composition	R-O
CP4 issuer-independent meaning of terms	R-M
CP5 per-assertion verification after combination, with linked provenance, version and status	R-P addresses verification
CP6 shared nodes and cycles (ownership, supply networks, delegation)	R-C
CP7 attested authority at the level the relying party requires	R-A
CP8 cost at scale (work per consumed fact, latency, throughput)	T2
CP9 longevity (checkable for the product's life)	—
CP10 jurisdictional interoperability	C2

Relational facts, shared meaning, provenance and authority recur across the examples, sometimes as design assumptions. Shared identifiers, order-independent composition and shared or cyclic structure matter for the multi-issuer scenarios in which they are used. The formal conditions in Section 4.4 describe those scenarios.

4 Formal model

4.1 Preliminaries

Let $\mathcal{I}$, $\mathcal{B}$, $\mathcal{L}$ be IRIs, blank nodes and literals. A statement is a triple $t = (s, p, o) \in (\mathcal{I} \cup \mathcal{B}) \times \mathcal{I} \times (\mathcal{I} \cup \mathcal{B} \cup \mathcal{L})$. A graph is a finite set of triples, and a dataset has a default graph and may have named graphs [8]. For signing we use its set of quads, with the graph position empty for a default-graph triple. Empty named graphs contribute no quads. Write $D \cong D'$ when two quad sets are isomorphic under one consistent blank-node renaming. A *provenance-preserving merge* stores each source dataset under a distinct snapshot identifier and preserves its original graph positions. It standardises blank nodes apart between snapshots and preserves their identity

within each snapshot. Under RDF simple semantics, isomorphic graphs have the same meaning and entailment is monotone [9].

Meaning of a document. For a JSON-LD document J and a context-resolution function ρ, $E_\rho(J)$ is the dataset obtained by expansion and conversion to RDF [12]. The meaning of J under ρ is $E_\rho(J)$ up to $\cong$.

Canonicalisation. Fix an RDFC-1.0 implementation, canonicalisation options and input normalisation. For admitted quad sets, RDFC-1.0 produces the same canonical output exactly for isomorphic inputs when canonicalisation succeeds [23]. **Assumption A1:** each admitted source dataset is within the profile's resource limits and has either no blank nodes or blank nodes distinguished by their first-degree hashes. This avoids the n-degree search and leaves hashing and sorting work. A1 applies to the resulting quads of each source snapshot. First-degree hashes can coincide in the merged dataset. JSON-LD lists can introduce blank nodes even when explicit entities have IRIs. A verifier enforcing A1 checks this condition directly. A work limit that lets some n-degree searches complete does not enforce it.

A profile also needs resource controls. Fujie and Suzuki show that input symmetry can make small graphs expensive and that implementations differ in how they limit work (§V-C) [68]. Section 5.5 reports a separate test of rdf-canonize 4.0.1 with an input-dependent work budget. It measures the behaviour of that control. The benchmark uses the resource limit and does not implement an A1 admission check.

Cryptography. H is collision-resistant, and the signature scheme is EUF-CMA secure. The tested signatures and this analysis use classical assumptions; the paper does not prove the security of a post-quantum hybrid construction.

4.2 Objects

Credential and snapshot. $c = (id_c, i_c, D_c, \pi_c)$ has an identifier, issuer, dataset and proof. The storage profile assigns an immutable snapshot identifier to each version and binds it to the source graph positions, proof, suite or envelope profile and verification inputs. It retains the suite-specific proof configuration and any source artefact required for verification. *Digest link.* A statement in D_c identifies the target snapshot and its digest algorithm and commits to a length-delimited encoding of the target profile identifier and the profile's prepared input $\mathrm{Prep}(c')$ (Section 4.3). The profile must state which inputs it prepares. The digest checks the prepared content. A profile that commits only to content must check the referenced snapshot and its issuer proof separately. The relying party separately verifies the proof, issuer, scope, validity and status. Digest links form a directed graph. The chain in the EBW proposal motivates this design but does not prescribe digest links.

Verification dependency: for a fact t the relying party's policy defines a rooted, finite dependency graph whose nodes are credentials and whose edges are the authority and status checks the policy requires (Section 5.6). The policy also defines the validity time, the scope of each authorisation, whether several authorisations are required jointly (AND) or alternatively (OR), and a rule for cycles (a cycle is broken at an anchor source or the fact is rejected). We write $\mathrm{Auth}(t)$ for the set of authority facts the policy requires directly for t.

Assurance levels. $\Lambda = \{\bot < U < E < Q\}$, where $\bot$ has no attributable source, U is accepted on unsigned local provenance, E is based on a verified issuer proof without qualification, and Q is based on a validated QEAA or PuB-EAA within the scope of the model (Section 2.4). λ_{req} is the policy's minimum for a slot. Technical verification and the policy's decision to accept an input are separate checks.

Admissible profiles. Adm_Q is the set of issuance profiles eligible for QEAA or PuB-EAA under the applicable EUDI Wallet rules. In the current baseline the relevant profiles are SD-JWT VC and mdoc under TS 119 472-1 clauses 5 and 6 [2–4]. This is a legal eligibility premise. Eligibility applies to an issuance profile and the relevant type of attestation.

Pluggable anchor sources. The policy gives each anchor source S a validation scope and a ceiling $\ell(S) \in \Lambda$ for what it establishes. A national trusted list and certificate path can identify a listed service. The verifier still checks eligibility of the attestation, including the applicable issuer category and service scope. A qualified electronic ledger may establish integrity and ordering for a status record. Ecosystem registries, for example those of EBSI [69] or UNTP, establish membership under their rules. A recognised third-country source has the scope and legal effect given by the applicable recognition rules. DID resolution supplies verification methods, while authority depends on the policy's binding of those methods to the issuer and its scope. On accepted, verified evidence, let $\lambda(c)$ denote the validated level. A credential outside Adm_Q has $\lambda(c) \leq E$ in this model. Profile eligibility is one condition for assigning Q, alongside the applicable issuer, service, proof, scope, validity, status and recognition checks. When a required condition fails, the policy can reject the input or take an expressly permitted fallback.

Attribute-set model $\mathcal{F}$. This is an abstract profile in which each finite document presents claims about one implicit subject. Claims are scoped by the credential type and the profile provides neither shared identifiers for other entities nor a cross-document predicate mapping or fact-union operation. JSON has a tree syntax [70]. SD-JWT VC and mdoc provide type-scoped claim structures in JSON and CBOR respectively. A deployment can restrict them to $\mathcal{F}$ or add shared identifiers and a predicate mapping. T1(b) concerns the restricted profile.

Policies and decisions: a trust policy assigns each *slot* (the role a fact plays in a decision) acceptable issuer classes, scope and time constraints, and a required level λ_{req}. A decision procedure Π is a deterministic function from the set of accepted facts to outcomes. Missing, stale or rejected evidence yields a defined outcome (reject or escalate), which the policy specifies.

Derived facts and re-signing (used in Sections 5.2 and 5.6). A relying party can issue its own assertion about a computation or interpretation. Its proof authenticates that assertion at the level the policy validates for it. A link to the original evidence can preserve source attribution if the relying party can check the original proof and the mapping. Re-serialising the same verified RDF dataset under a compatible profile does not create a new assertion.

4.3 Proof domains

A securing mechanism is $\mathsf{M} = (\mathrm{Prep}, \mathsf{Sign}, \mathsf{Verify})$, where Prep maps a representation under fixed proof options to the signed message. Its *proof domain* is the equivalence $r \sim_{\mathsf{M}} r' \iff \mathrm{Prep}(r) = \mathrm{Prep}(r')$. With the same key and verification inputs, equivalent representations yield the same cryptographic signature check. Under EUF-CMA, successful adversarial use of an issuer signature for a distinct message that the issuer did not sign is negligible. Acceptance can also depend on proof format, policy and status checks.

Domain	Mechanisms	Invariant under
Bytes	JWS, SD-JWT, SD-JWT VC, VC-JOSE-COSE, COSE (mdoc)	changes outside the signed byte string only (transport framing, unsigned headers)
JSON value	Data Integrity, JCS cryptosuites	whitespace, key order
Graph	Data Integrity, RDFC cryptosuites	any serialisation of an isomorphic quad set (N-Quads, TriG, any JSON-LD form, a triple store)

A boundary case is a JWS or COSE signature over a graph digest. Its representation includes an envelope, but the authenticated content is a commitment to $\mathrm{C}(D)$. Whether a verifier can reconstruct the signing input from its graph also depends on the retained protected headers and the profile's rules for serialising the digest.

Architecture A (graph-native; Figure 3): VCDM 2.0 with a graph-domain Data Integrity proof. For L2(a) the profile uses a full-credential RDF signature with recoverable proof configuration. Consumers preserve the signed quad set, graph positions and proof data for each snapshot.

Architecture B (nested hybrid): JSON-LD inside a bytes-domain envelope. B1 carries JSON-LD as claims in an IETF SD-JWT VC with vct and @context. Eligibility of any B1 deployment depends on the applicable issuance profile. B2 uses W3C VC-JOSE-COSE, a distinct profile [24]. UNTP 0.7.0 is an example of the JOSE variant [39]. In the comparison below, B uses the envelope as its issuer proof and does not require an inner Data Integrity proof. A construction retaining both proofs is a separate composition; Appendix C supplies a complete example.

Definition 4.1 (verification path, homogeneous path, single-pipeline objective). A *verification path* for t is the rooted dependency graph the relying party checks from t to the anchor sources. It includes the credential stating t, the authority facts $\mathrm{Auth}(t)$ and their dependencies. The path is *homogeneous* when every credential follows the same verification profile and proof domain, so one implementation of that profile can check the path. The profile specifies the permitted parsing, proof, key and status checks. A relying party adopts the *single-pipeline objective* if it requires homogeneous paths, checks them at its own point of use and does not delegate their verification to a separately trusted verifier. A path can be homogeneous while mixing validated levels E and Q.

4.4 Requirements as conditions

For a representation scheme $\mathcal{R} = (\mathrm{Doc}, \oplus, [\![\cdot]\!])$ with documents, a combination operator and a semantics mapping each document to a finite set of atomic facts over a universe of entities, literal values and predicate names:

- **R-G:** $[\![\cdot]\!]$ expresses facts of any arity ≥ 2, with typed argument positions (entity or literal), whose entity arguments may be described in other documents.
- **R-I:** an entity reference that occurs in documents of different issuers denotes the same entity in all of them. Entities referenced in one document only may be local to it (existential scope).
- **R-M:** a predicate name's denotation does not depend on the issuer.

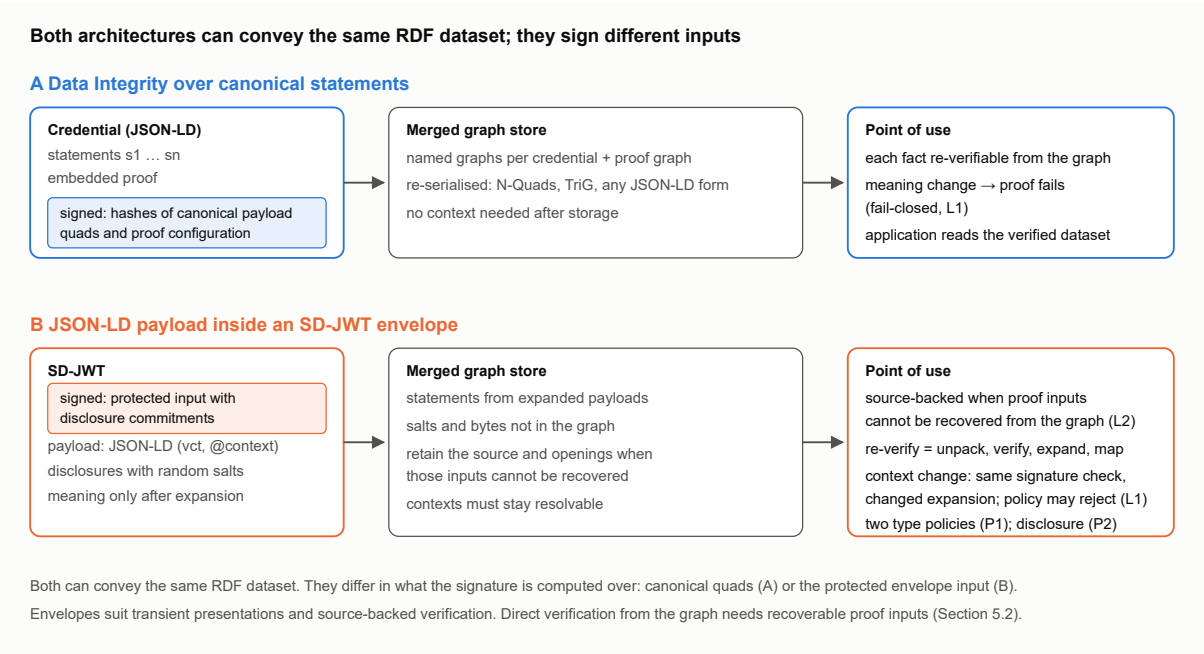


Figure 3: The architectures can convey the same RDF dataset. A checks a proof over the canonical quads and the canonical proof configuration. B checks a proof over the protected envelope input. In the erasure case of L2(b), B needs retained source evidence for direct verification after a merge.

- **R-O:** $\oplus$ is commutative, associative and idempotent, and $[\![d_1 \oplus d_2]\!] = [\![d_1]\!] \cup [\![d_2]\!]$.
- **R-C:** the scheme represents finite structures in which an entity is the argument of several facts (shared node) and in which facts form directed cycles, without duplicating the entity.
- **R-P:** per-fact verifiability after merge (Definition 5.1).
- **R-A:** attainable assurance (Section 5.6).

5 Results

5.1 An RDF encoding of the requirements

Theorem T1 (graph encoding). (a) Let $\mathcal{R}$ satisfy R-G, R-I, R-M and R-O, with globally named entities and predicates encoded by distinct IRIs in their respective roles, and literal values encoded injectively as RDF literals. Document-local entities have persistent source scope when fact sets are combined. There is an encoding enc of finite fact sets into finite RDF triple sets that is injective up to renaming of local entities and satisfies $\mathrm{enc}([\![d_1 \oplus d_2]\!]) \cong \mathrm{enc}([\![d_1]\!]) \cup \mathrm{enc}([\![d_2]\!])$, with compatible blank-node identities on the right. It represents shared nodes and directed cycles (R-C). (b) The restricted attribute-set profile $\mathcal{F}$ provides no cross-document identity for referenced entities, common predicate mapping or fact-union operation. A deployment can supply these and then use the encoding of (a).

Globally keyed relational tables with typed columns and set union also represent these fact sets. RDF supplies an encoding with the standardised semantics, JSON syntax and canonicalisation described in Section 2.

Proof. (a) Reserve encoding IRIs disjoint from the entity and predicate vocabularies. Encode a binary atom with an entity in first position as an ordinary triple. For each other atom

$r(e_1, \ldots, e_n)$ use a node x with $(x, \texttt{enc:pred}, r)$ and $(x, \texttt{enc:arg}_j, e_j)$ for $1 \leq j \leq n$. For a grounded atom, use as x an IRI derived from an injective, reversible encoding of the predicate, arity and typed argument tuple, for example a base64url representation of a length-delimited byte encoding in the reserved namespace. For an atom with a local entity, allocate a blank node indexed by the scoped atom, distinct from every blank node allocated to an entity. A local entity has the same blank node in all the triples representing it. The reserved predicates identify reified atoms, which are decoded by reading their indexed arguments. All other triples decode as binary atoms. This recovers the facts up to renaming of local entities. For union, use the same identifiers for the same scoped entities and atoms. Separate unrelated source scopes, and keep the same scope when the same source snapshot is encountered twice. Encoding each atom then commutes with set union. The binary facts $p(a, b)$ and $p(b, a)$ yield (a, p, b) and (b, p, a), sharing a and b and forming a directed cycle.

(b) Two profiles can use status for different properties. A fact-level union needs a mapping to issuer-independent predicates, for example agreed IRIs that distinguish register status from portal-account status. A nested object without a reference cannot distinguish an entity shared by two documents from two distinct entities with identical properties. A raw object merge can overwrite a colliding member and depend on input order. These are limitations of the specified flat profile and merge rule. Shared identifiers and a mapping allow a deployment to perform fact-level union and represent shared nodes and cycles with references.

T1 concerns the facts denoted by a document. JSON-LD supplies a JSON syntax for RDF. An $\mathcal{F}$ deployment can also denote a graph once it adds shared identifiers and a predicate mapping. Section 5.2 considers what the proof authenticates.

5.2 What the proof covers

The two verification modes differ in what the relying party retains after a merge.

Definition 5.1 (source-backed and graph-native verification). Let M be a provenance-preserving merged dataset with the snapshot associations, original graph positions and proof configuration of Section 4.2 recorded in the graph store. A statement is *source-backed verifiable* if the relying party can verify the retained issuer artefact and any required openings, validate its interpretation under an authorised mapping and establish that the statement occurs in that interpretation at the recorded source and graph position. A mechanism has *graph-native per-fact verifiability* (PFV) in this model if an efficient verifier, given M, a statement, the designated source snapshot and original graph position, and the policy and trust material, can establish membership in a successfully verified source snapshot without retained source bytes, disclosure openings or auxiliary source-reconstruction data. For a blank-node statement, membership is scoped to that snapshot.

Lemma L2. (a) Architecture A has graph-native PFV for a full-credential RDF signature when each admitted snapshot satisfies A1 and the stored configuration permits recovery of the suite's signing input. (b) For ordinary direct verification of a bytes-domain signature, the verifier needs its exact protected input and any openings needed for the claim. If these are erased and cannot be recovered efficiently from what remains, that route is unavailable. (c) For B in this erasure case, retaining and authenticating the source artefact with the required disclosures and validating its RDF interpretation supplies a source-backed route. Reusing an authenticated interpretation requires an explicit policy for checking its source binding and freshness.

Proof. (a) Retrieve the source snapshot containing the statement. Restore the original graph positions and proof configuration. This recovers the issuer's quad set up to a consistent blank-node renaming. By A1, canonicalisation succeeds and produces the issuer's canonical quads. For eddsa-rdfc-2022 (§3.2) [6], the message comprises the concatenated hashes of the canonical proof

configuration and the canonical unsecured document. A verifier that reconstructs both inputs can verify the proof and check the statement's source and graph position. Completeness follows from the reconstructed inputs and signature correctness. Substitution of an unsigned message would require a hash collision or a signature forgery under the cryptographic assumptions. (b) The standard verification call takes the protected message as input, and selective claims also require an opening that matches the signed commitment. A profile can fix a deterministic reconstruction, or allow a verifier to try a small set of candidate protected inputs. Neither route works where the necessary inputs are unavailable and infeasible to recover. For a selectively disclosed SD-JWT member, the issuer signs a commitment and the disclosure carries the salt. RFC 9901 recommends a 128-bit random portion (§9.3) [28]. The recommendation alone does not prove an erasure bound for every profile, and cleartext claims need no opening. (c) Verifying the retained artefact and openings authenticates its claims. Applying the authorised mapping and checking membership establishes the recorded statement's relationship to those claims.

Source-backed verification serves a different purpose from graph-native reconstruction. OCI illustrates another use of envelopes. It specifies embedded linked-data proofs for credentials and a JWT presentation for product-identifier verification [48, 49].

L2 also bears on how qualified and self-asserted facts combine. Suppose the authority edges that support a fact are issued in an attribute-set format while the fact itself is a graph-domain credential. L3 describes ways to retain a verifiable connection to the issuer.

Lemma L3 (authority edges across a format boundary). Let t be stated in a graph-domain credential and let its authority fact $a \in \mathrm{Auth}(t)$ originate in a bytes-domain credential whose verification input is unavailable in the merged graph under L2(b). A projection that carries neither the issuer's proof nor other verifiable evidence of its relation to the source cannot establish issuer authentication from that graph alone. Available routes include (i) verifying the retained artefact and validating the mapping, (ii) checking a separately trusted verifier's authenticated result, and (iii) obtaining the issuer's authentication of the graph representation. Route (ii) introduces reliance on the delegated verifier. Route (iii) can form a homogeneous path if the graph credential follows the agreed profile.

Proof. In (i), the original proof authenticates the source claims. A validated mapping, which may itself be bound by the issuer or the profile, establishes which statements those claims support. In (ii), the delegate vouches for a verification result. The relying party must check the delegate's authority and bind its result to the source snapshot, mapping version, decision input and validity conditions [71]. In (iii), the issuer supplies evidence of the RDF claim directly, for example by dual issuance or by authenticating a graph commitment under a suitable profile. A projection without any such evidence fails the source and membership checks of Definition 5.1 under L2(b).

A relying party that has adopted the single-pipeline objective (Definition 4.1) excludes a different envelope profile and separately trusted delegated verification on its own path. It can still consider other designs, such as a bridge proof that establishes the relation to the original input, if the adopted profile and applicable issuance rules support it. A cache of a verified source-backed view is a variant of (i). Its policy specifies how the source, mapping, freshness and status remain bound to the cached result.

Lemma L1 (context changes). Let J reference an unpinned context and suppose resolution changes from ρ to ρ' so that both expansions succeed and $E_{\rho'}(J) \ncong E_{\rho}(J)$. (a) B's signature check on the unchanged artefact and disclosure set is unaffected. A consumer that expands under ρ' obtains the changed meaning, and acceptance depends on its profile and policy. (b) For A with a full-credential RDF signature under a fixed processing profile, the old proof fails to authenticate a changed, otherwise unsigned canonical input except with negligible probability under the cryptographic assumptions. A verifier reconstructing the signed quad set and proof configuration from storage need not resolve a context to canonicalise those quads. (c) Fixing

and authenticating the complete context closure and fixing the base IRI and processing settings fixes the meaning for a given authenticated disclosure set. P2 considers a change in that set.

Proof. (a) The protected envelope input and disclosure openings remain unchanged. (b) On admitted inputs A1 gives successful canonicalisation. The changed quad set has a distinct canonical form, so its hash differs under the collision-resistance assumption and the existing signature cannot authenticate the changed, otherwise unsigned message under the unforgeability assumption. A processing error causes rejection. From storage the verifier checks the reconstructed inputs as in L2(a). (c) The fixed settings and authenticated contexts make expansion deterministic for the reconstructed document.

The same holds for terms that a context change leaves undefined: generic JSON-LD processing drops them [12], so in B a context change that un-defines a term removes facts under a valid signature, whereas a Data Integrity implementation must raise an error when securing would lose data (§2.4.3) [5] and verification of the changed input fails as in (b).

Proposition P1 (two type systems). In B1, vct identifies a credential type and @type can name RDF classes. Separate application policies on these types can disagree on the same authenticated payload. *Witness.* An application schema selected by vct accepts a numeric value without a unit. A graph application applies a SHACL shape that targets the RDF class ex:Limit and requires ex:unit. Consider the reconstructed application-claims fragment (the signed envelope and issuer metadata are omitted):

```
{
  "@context": {
    "ex": "https://example.org/",
    "value": "ex:value",
    "unit": "ex:unit",
    "vct": {
      "@id": "ex:credentialType",
      "@type": "@id"
    }
  },
  "@id": "ex:o",
  "@type": "ex:Limit",
  "vct": "https://example.org/limit-profile",
  "value": 5
}
```

The explicit mapping of `vct` to `ex:credentialType` preserves it as an IRI-valued RDF property. This is an example-specific mapping, not a standardized SD-JWT vocabulary. Without it, ordinary JSON-LD processing would drop `vct`, and lossless processing would reject the input. The selected schema accepts the payload, while the shape targeting https://example.org/Limit rejects the RDF view because the subject has no ex:unit. Both policies operate on the same authenticated claims. The profile decides which checks govern use and what to do if they disagree [29, 72]. The same policy mismatch can arise in a plain VCDM credential using `credentialSchema` and RDF types; VCDM permits JSON Schema validation [22]. B1 must additionally define how `vct` and RDF typing select compatible rules. Selecting a graph-proof format does not remove the need to reconcile application validation policies.

Proposition P2 (disclosure soundness). A presentation is disclosure-sound if the quads obtained from its verified issuer payload form a subset of the issuance quad set under one injective blank-node correspondence. For an RDF proof, that set is the issuer-authenticated payload. For an envelope, it is the complete committed payload expanded under the issuance

profile. The comparison excludes presentation metadata and inferred quads. (a) The two RDF selective-disclosure suites discussed below are disclosure-sound under their cryptographic soundness assumptions when verification succeeds, the profile checks injectivity of the blank-node relabelling where present, and the application uses the verified payload. (b) A bytes proof with JSON-LD 1.1 disclosure can verify a presentation whose expansion fails this condition.

Proof. (a) For ecdsa-sd-2023, the verifier reconstructs suite-labelled quads, checks the issuer-authenticated mandatory group and checks each revealed non-mandatory quad against a signature under the proof-scoped key authenticated by the base proof [26]. For bbs-2023, the verifier reconstructs the suite-labelled disclosed messages and verifies their inclusion under the issuer's proof, with the mandatory quads bound in the proof's header [27]. Subject to suite soundness, these checks authenticate the revealed quads with the suite's blank-node labelling. (b) Consider the following application-claims fragment

```
{
  "@context": {
    "@version": 1.1,
    "ex": "https://example.org/",
    "id": "@id",
    "value": "ex:v2",
    "ex:Limit": {
      "@id": "https://example.org/Limit",
      "@context": {
        "value": "ex:v1"
      }
    }
  },
  "id": "ex:o",
  "@type": "ex:Limit",
  "value": 5
}
```

in which the term ex:Limit carries a type-scoped context (§4.1.8) [12]. An illustrative ES256 SD-JWT carries authenticated `iss` and `vct` metadata and commits to the RFC 9901 disclosure of `@type`, with the context, identifier and value clear. After signature and disclosure verification, an explicit application projection retains the issuer/type metadata separately and expands only the shown claims; it does not silently drop undefined metadata during JSON-LD processing. Both presentations, with and without that disclosure, pass the same issuer-signature and disclosure-digest checks. This cryptographic result does not by itself establish acceptance under a credential profile. With jsonld.js 8.3.3, full disclosure expands to ex:o rdf:type ex:Limit and ex:o ex:v1 "5"^^xsd:integer. Withholding @type yields ex:o ex:v2 "5"^^xsd:integer. The pinned context is the same. This witness applies when the application profile accepts both presentations. It does not establish conformance with an EUDI or W3C wallet profile.

A profile can keep interpretation-critical members clear or check that each allowed disclosure produces only issuer-authenticated quads. Pinning the context without such a rule leaves the example unchanged.

5.3 Selective disclosure

The formats differ in what a holder can disclose and what the verifier checks. The comparison below separates these properties from the full-disclosure workload used in Section 5.5.

What VCDM 2.0 offers. Data Integrity specifies ecdsa-sd-2023 [26] and bbs-2023 [27]. The first authenticates mandatory quads and a proof-scoped verification key with the base signature and signs the non-mandatory quads with the corresponding private key. The second derives a proof for disclosed messages of an issuer-signed credential with the mandatory group bound in the signed header. In both, JSON Pointer selection and structural dependencies can reveal groups of quads. VC-JOSE-COSE also specifies selective disclosure at JSON member or array-element level [24, 28].

Proposition P3 (granularity, verification and linkability). (a) The sets a holder can disclose depend on the format's selection rules, mandatory groups and the input structure. A single disclosable JSON member can expand to several quads, and A's pointer selection can also reveal groups. (b) For the named suites, a verifier checks the SD-JWT issuer signature and the commitments for disclosed members or elements, with an additional key-binding proof where required. It checks the ecdsa-sd-2023 base signature and one signature per revealed non-mandatory quad, or the derived BBS proof and its disclosed messages. These are counts of operations; Section 5.5 reports measured times for two published implementations. (c) BBS supports unlinkable derived proofs under its privacy conditions. Two presentations of the same SD-JWT can be linked through the repeated issuer-signed JWT. Using different, single-use credentials can mitigate that channel. Stable identifiers, values, status references and holder metadata can still link presentations. *Proof.* (a) follows from the selection and grouping rules. (b) follows from the verification algorithms. (c) follows from the derived-proof and privacy properties stated in the specifications [26–28].

Proposition P5 (composing disclosure mechanisms). Consider a profile that requires a verifier to rely on both SD-JWT disclosures and a Data Integrity selective-disclosure proof in the embedded credential. (a) The profile must specify both proof scopes, the order of reconstruction and verification, and the outcome if the proofs authenticate different inputs. (b) If both layers supply holder binding, the profile must specify how their keys, audience and challenge checks relate. (c) For a requirement limited to authenticated RDF disclosures and holder binding already supported by an A profile, an extra envelope disclosure mechanism is unnecessary. *Proof.* (a) The outer proof and openings authenticate JSON claims and the inner proof authenticates the disclosed quads. The scope of one check does not determine the scope of the other (P2). (b) The distinct binding mechanisms authenticate their respective presentation inputs [22, 28]. (c) A presentation meeting the stipulated requirement exists by the premise. This gives no equivalence claim for wallet or protocol requirements outside that premise.

Workload assumption. Section 6 models full-record access and repeated verification. ESPR specifies access rules for DPP data, the Product Liability Directive provides for proportionate disclosure in litigation, and the AI Act has logging provisions for applicable high-risk systems [57, 59, 64]. How often a verifier reads a record in full is a scenario assumption. T2 addresses this full-disclosure workload. Selective disclosure remains relevant to confidential bills of materials and personal data in mandates. A verifier that must check a threshold without seeing the value needs an issuer-signed predicate or a suitable proof. Wallet profiles can provide selective disclosure and holder binding. Section 5.5 compares two implementations under their tested disclosure policies.

5.4 Complexity as a security property

Security engineering has long treated complexity as a source of insecurity [73, 74]. Differential parsing can make a verification component and an application act on different inputs [75, 76]. The history of JWT algorithm confusion and the associated validation guidance give examples of why profiles must specify the intended checks [77–79].

Definition 5.2 (use-bound). Verification is *use-bound* if the application consumes the authenticated view or a specified derivation of it. The derivation uses only the authenticated input and the interpretation parameters fixed by the admitted profile.

Proposition P4. (a) A verifier for A is use-bound when the application reads the verified RDF dataset or a specified derivation. An application that reads raw JSON array order for a property whose RDF representation is unordered can distinguish two equivalent signed inputs. (b) B can be use-bound when the profile fixes reconstruction and the interpretation used by the application, including the inputs to expansion and the applicable type checks. (c) For a fixed finite test suite, agreement on the tests alone establishes agreement on those tested cases. A profile should define the admitted interpretation and check it with conformance and differential tests.

Proof. (a) The authenticated RDF view determines the stipulated application input. Permuting the values of an ordinary multivalued JSON-LD property leaves the quads unchanged but can change what a raw JSON reader finds at index zero. (b) The authenticated reconstruction and fixed interpretation determine the application view. If resolution or reconstruction can change the view outside the admitted rules, L1 gives an example of the resulting gap. Separate type policies can also disagree, as in P1. A deterministic reconstruction may still produce a graph that fails the subset condition in P2. Use-binding alone does not establish disclosure soundness. (c) An untested input can distinguish implementations unless additional constraints establish agreement on the admitted domain.

Generic JSON-LD processing can discard undefined terms and relative URLs. Data Integrity requires an error when the transformation would lose data (§2.4.3) [5]. A 2024 Data Integrity issue raised concerns about term redefinition and the treatment of contexts during the proof transform [80]. In their tests of a JSON-LD/RDFC pipeline, Fujie and Suzuki measured the cost of poison graphs, showed that context substitution changed the expanded IRI and canonical quads, which the old RDF proof cannot authenticate, and observed a malicious URL at the document-loader boundary without sending the request. The JSON-LD pipeline used library defaults without context validation. Their separate SD-JWT VC and mdoc implementations rejected the respective JOSE and tampering cases tested (§§IV-C, V-A and VI-A) [68]. These are results for specified configurations and attacks.

A local loader can refuse unrecognised URLs and remove remote context retrieval. M2 uses fixed local contexts, while A1 restricts the quad sets. These controls have different purposes. A verifier can screen claimed issuer identifiers, proof structure and input size before canonicalisation. An invalid proof naming an allowed issuer can still reach the expensive step. A bytes-domain verifier can check the envelope signature before expanding JSON-LD. The profile needs a work budget, size limits and an execution limit that interrupts the work it bounds [68]. A graph verifier checking admitted immutable snapshots works with a narrower input class than it does at ingestion, provided the store preserves the admitted datasets and their integrity. A verifier refreshing a JSON-LD mapping repeats expansion. With a fixed local loader it does so without network context retrieval. In both designs the profile specifies what the application may rely on.

5.5 Cost at scale

Primitives. For a credential of n quads, $x(n)$ is JSON-LD expansion and conversion to RDF, $\kappa(n)$ is canonicalisation of its parsed dataset, $h(n)$ is hashing of the canonical form, and g is canonicalisation and hashing of the parsed proof options. The cost $q(n)$ covers retrieval and extraction of the source quads. Let p be envelope processing, including one signature verification, and let v and s be one signature verification and one signature generation. The event costs in T2 are issuance (I_A, I_B), ingestion (A_0, B_0) and subsequent verification (A_1, B_1). The store path

for A retrieves and verifies both the credential and its proof options. Its cost includes q, κ, h, g, v and retrieval of the proof options and signature. The measured q primitive covers the credential dataset alone. Issuance and ingestion in A also expand the document and proof options. B retrieves the envelope, processes it and computes a fresh RDF view ($B_1 \approx p + x$), or checks just the envelope ($B_1^s \approx p$). The latter leaves any cached view and its binding to the source outside the timed path. These are schematic costs. The benchmark times each path as one call.

Workload. A credential is issued, ingested once, and verified R more times. R counts the independent verifications the relying parties choose to perform, for example at a trust boundary, at points of use or at audits. *Modelling assumptions.* M1 uses the same verification schedule for both architectures. A re-verifies the stored quads, and B reprocesses the retained envelope, with a fresh expansion in the primary comparison. The envelope-only comparison assumes a previously verified view and a trusted binding to the retained artefact and the applicable interpretation profile. Its verification time excludes retrieving or validating that binding and view. M2 gives both paths the same immutable context snapshot and local loader policy. A1 applies to the admitted source datasets. Costs add across events and, when aggregating, across credentials.

Theorem T2. Let $C_A(R) = I_A + A_0 + R\,A_1$ and $C_B(R) = I_B + B_0 + R\,B_1$ be the total cost of issuing, ingesting and re-verifying one credential R times. Let $\Delta_0 = (I_A - I_B) + (A_0 - B_0)$ be A's extra up-front cost and $\delta = B_1 - A_1$ its saving per re-verification. (i) If $\delta > 0$, A costs less for $R > R^* = \Delta_0/\delta$, B costs less for $R < R^*$, and they tie at R^* if it is in the workload. The first non-negative integer at which A costs less is $\max(0, \lfloor R^* \rfloor + 1)$. (ii) If $\delta \leq 0$ and $\Delta_0 > 0$, B costs less for every $R \geq 0$. (iii) For a fact base of N credentials with constants $\Delta_{0,c}$, δ_c and R_c re-verifications each, A has lower total work iff $\sum_c (R_c\delta_c - \Delta_{0,c}) > 0$. With common constants and $\delta > 0$ this is $\bar{R} > R^*$ for the mean $\bar{R}$. The same sign holds per consumed fact under a common positive denominator. (iv) Excluding issuance, the equality occurs at $R_v^* = (A_0 - B_0)/\delta$ for $\delta \neq 0$ when that value is in the workload. The direction of preference follows the sign of δ. For $\delta = 0$ only the up-front difference matters.

Proof. $C_B(R) - C_A(R) = (I_B - I_A) + (B_0 - A_0) + R(B_1 - A_1) = R\delta - \Delta_0$. (i) For $\delta > 0$ the difference is positive above the root, negative below it and zero at it. The integer threshold follows from the strict inequality. (ii) For $\delta \leq 0$ and $R \geq 0$ the difference is at most $-\Delta_0 < 0$. (iii) Sum the per-credential differences. Division by a common positive denominator preserves the sign. (iv) Drop $I_A - I_B$ from Δ_0 and apply the same comparison.

Hypothesis H1. For the specified profile and verification paths, processing and expanding the retained envelope costs more per re-verification than recovering and verifying the stored source quads, so $\delta > 0$. The benchmark measures these paths under the conditions below.

Benchmark (run 3). The prototypes start with the same synthetic JSON-LD credential documents, with typed literals and components, 17 to 574 quads, and IRI-only and blank-node versions. They use one Ed25519 key derived from a fixed seed. Each size and shape is tested separately with 500 credentials in an in-memory N3.js 2.13.12 store [81]. The synthetic credential graphs originate as default graphs. A stores the credential and proof-options quads under separate graph names and keeps the signature in a map. B uses the same credential graph name when inserting its expanded payload and retains the token in a map. A_1 retrieves both graphs, returns their quads to the default graph, canonicalises and hashes both, and verifies the signature. B_1 retrieves and processes the token, re-expands the payload and checks that the resulting dataset is non-empty. Comparison to any stored view is outside the path. B_1^s retrieves and processes the token and checks its identifier. For the first credential of each size and shape, the experiment checked A's signature against the co-populated store and compared the canonical dataset with B's fresh expansion. The prototype envelope verifier rejects repeated supplied disclosures and unused disclosures, within the benchmark fixtures.

Timing and comparison libraries. We used jsonld.js 8.3.3, rdf-canonize 4.0.1, Node.js 24.21.0 and local pinned contexts on an AMD Ryzen AI 9 365. We recorded elapsed time on one thread using process.hrtime.bigint. Each batch-based single-thread measurement uses 10 warm-up calls and 11 batches of at least 200 ms, with medians computed from the archived batch means. A separate latency sample of 2,000 calls per path follows 20 warm-up calls. Repeating the signature-verification timing changed its median by at most 6 %. The benchmark times issuance, ingestion into a fresh store and each re-verification path separately. Network access and production trust checks are outside the paths. Reference-library calls on the IRI-only credentials use the same Ed25519 key [82, 83]. The reference A signing times are 4–14 % higher than the prototype's. Its verification call has a different scope from A_0, which also inserts quads. The reference B verification call on a separately issued token takes 3.0–6.7 times the prototype envelope-processing time. With expansion, the ratio to the prototype B_1 ranges from 1.2 to 3.6. These comparisons describe the timed calls. Appendix B describes the paths, code and results.

Table 3: Prototype timings in microseconds per credential for IRI-only inputs in run 3. A_1, B_1 and B_1^s are medians of batch means from individual calls to the paths above. $\delta = B_1 - A_1$. The crossovers follow T2 from the measured costs, using B_1 with a fresh expansion. Values are rounded directly from the archived data.

n	x	κ	q	p	A_1	B_1	B_1^s	δ	R^* **incl. issuance**	R^* **consumers only**
17	534	7	9	91	107	704	92	597	4.8	2.0
50	633	21	27	130	149	797	134	649	4.6	1.8
146	804	58	100	249	282	1,063	254	781	4.2	1.6
374	1,086	158	196	662	563	1,884	652	1,321	2.1	0.6
574	1,452	253	262	1,188	766	2,760	1,193	1,994	1.3	0.3

Results. With fresh expansion for B, $\delta > 0$ at all tested sizes and for both shapes. For IRI-only credentials the difference grows from 0.60 ms at 17 quads to 1.99 ms at 574. The crossover falls from 4.84 to 1.32 additional verifications, with 4.24 at 146 quads. Excluding issuance it falls from 1.97 to 0.32. With blank nodes the crossover ranges from 1.71 to 5.39. Blank-node components increased canonicalisation cost by about 58–150 % and A_1 by about 6–36 % in these fixtures. For the three IRI-only sizes below 150 quads, the envelope-only check cost about 10–14 % less than A_1. It cost about 16 % and 56 % more at 374 and 574 quads. The corresponding modelled crossovers against the envelope-only path are about 31 and 6.1 additional verifications. With blank nodes, the envelope-only path cost 12–29 % less at the first four sizes and 32 % more at 574 quads. This path leaves the graph mapping trusted, as in M1.

Comparison with run 2. Issuance and first ingestion together cost A 2.6–3.3 ms more than B for the IRI-only credentials, and up to 3.63 ms more with blank nodes. Retrieval and extraction of the credential quads took about 9–262 µs for the IRI-only rows. Across both shapes, the run-3 A_1 is 1.26–2.43 times a reconstruction of $\kappa + g + v$ from run-3 component timings, and 1.40–2.78 times the archived run-2 S1 estimate. The latter comparison also reflects changes between runs. For fresh expansion, run 3 gives a lower crossover than run 2 for the 17-quad IRI-only fixture and a higher crossover for every other fixture.

Latency and throughput. In the per-call samples the medians were about 1–19 % below the corresponding medians of batch means. For A_1 and B_1^s below 374 quads, the per-call 99th percentiles were within 1.7 times their per-call medians. For B_1 the ratio ranged from about 4.0 to 8.7 across the two shapes and five sizes. The experiment did not isolate the cause of the tails. For the 146-quad IRI-only fixture, the reciprocal batch medians yield about 3,500 credential checks per second for A, 940 for B with fresh expansion and 3,900 for B envelope-only, on one thread.

Separate one-shot results. In their Node.js cloud environment with an inline context, Fujie and Suzuki report 0.265 ms for mdoc, 0.298 ms for SD-JWT VC and 0.534 ms for VCDM with eddsa-rdfc-2022 for one issue–present–verify exchange disclosing 5 of 20 attributes. Their EdDSA presentation re-signs the subset. The timing order therefore concerns tested paths with different issuer-authentication guarantees. Their study does not measure graph storage, independent reuse or the nested hybrid [84].

Adversarial shapes. A separate stress test timed canonicalisation of prepared quads with rdf-canonize 4.0.1, adapting families from Fujie and Suzuki [68]. For the tested positive finite work factors, the library sets a budget of m^k deep iterations for m blank nodes without unique first-degree hashes. If the computed count exceeds JavaScript's safe integer range, the library uses an unbounded count. Each adversarial result is one observed call. At $k = 2$ the complete graphs of 5, 6 and 8 blank nodes and the three-regular graphs of 6, 8 and 10 nodes all aborted in at most 2.5 ms. The five-node complete graph took about 15 ms without the limit, about 1,800 times the 8 µs baseline for the 17-quad IRI-only credential. The eight-node complete graph took about 11 s at $k = 16$. The ten benchmark inputs completed at each tested setting. Of 65 distinct RDFC test-suite input files, 46 completed at $k = 1$ and 61 at $k = 2$. The latter setting aborted on three high-complexity evaluation inputs and one intentional negative input. The higher settings ran only the 33 low-complexity files. Completion here records whether the generic call returned. It does not establish conformance against the test manifest.

The 16- and 64-node rings completed at $k = 2$ in about 1.6 ms and 53 ms. The latter is about 6,700 times the local baseline. A fixed exponent alone does not establish an application deadline when the budget grows with the number of colliding nodes. The profile therefore needs an explicit admission rule for A1 and resource controls sized to the deployment. Size and deep-iteration bounds are candidates. The test does not select a bound or establish a latency guarantee. The 100-node acyclic tree also exceeded the $k = 2$ budget, after about 77 ms. The higher settings

were not run on that input. This shows an admission and cost trade-off even for an acyclic shape. Appendix B gives the settings and results.

Selective disclosure. The Digital Bazaar ecdsa-sd-2023 and OpenWallet Foundation SD-JWT/ES256 libraries use P-256 with separate generated keys [82,83]. The benchmark selects the same named properties from the subject of each fixture. The paths use different disclosure and proof structures. A retains mandatory issuer and validity fields, and B verifies a presentation with status checking disabled and without a key-binding proof. The benchmark does not compare the resulting RDF datasets for equality. For the 50-quad fixture with 20 disclosable subject properties, the benchmark selects 1, 5, 10 and 20 properties. Median batch means were 6.6 ms for the A base proof and 0.85 ms for B issuance, 5.1–6.2 ms for A derivation and 0.25–0.27 ms for B presentation, and 3.0–3.5 ms for A verification and 0.26–0.53 ms for B verification. B verification plus fresh RDF expansion took 0.95–1.3 ms. For the 146-quad fixture with 60 disclosable properties, A verification took 3.5–5.3 ms for the selections of 15, 30 and 60 properties and 12.6 ms for the one-property selection. B verification took 0.45–1.6 ms across all four selections. At 50 quads the derived A credentials were 981–3,003 bytes and the B presentations 2,337–3,335 bytes. The timings compare the specified implementations and exclude a matched wallet exchange.

For comparison, Fujie and Suzuki's Environment A reports 3.367 ms to issue ecdsa-sd-2023, 1.807 ms to derive a 5-of-20 presentation and 1.668 ms to verify it, against 0.147 ms for SD-JWT VC issuance and 0.141 ms for verification. These are separate configurations. T2 uses the fully disclosed workload measured here [84].

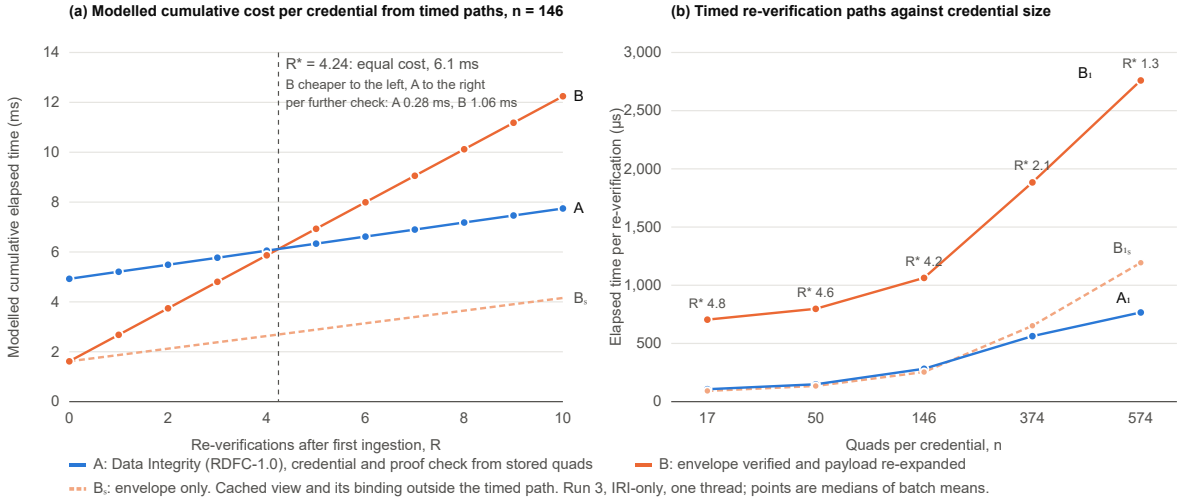


Figure 4: Cost model using run-3 prototype timings for IRI-only credentials on one thread. (a) The curves sum median batch means for issuance, ingestion and R re-verifications of a 146-quad credential. The calculated crossover is $R^* = 4.24$ for the envelope path with fresh expansion: at that point both paths have accumulated about 6.1 ms; left of it B costs less because its issuance and ingestion are cheaper, right of it A costs less because each further check costs 0.28 ms in A and 1.06 ms in B, so that at $R = 10$ B has cost 1.6 times A. The dashed envelope-only curve excludes checks on a cached RDF view. (b) Measured A_1, B_1 and B_1^s by credential size. The crossover labels use B_1 and include issuance.

Other schedules. In the tested full-disclosure workload, B costs less at issuance and ingestion together. Either architecture can reuse a verified view inside a trust domain, with freshness, integrity and status checks set by policy. If the independent re-verification schedules differ, the difference in T2 becomes $R_B B_1 - R_A A_1 - \Delta_0$ before any added cache costs. Replacing B_1 by B_1^s

prices the envelope check only. Key resolution, status checks, holder binding and transport were outside the compared paths. Fujie and Suzuki caution that these costs may outweigh the format differences in a wallet exchange [84]. L2 and L3 specify the evidence needed to authenticate the view.

5.6 Assurance propagation

Setting. A relying party accepts honest evidence S across a finite set of slots under its policy and obtains $o = \Pi(S)$. Each modelled slot has a baseline value and validated level in S. The policy gives each slot a required minimum λ_{req}, issuer class, scope and time constraints, and defines what happens on missing, stale or rejected evidence. An outcome-changing attack must induce a different accepted outcome $o' \neq o$. Rejection or escalation on its own is outside this attack objective.

Check level. For an accepted honest slot t, let $\chi(t)$ be the level of the evidence actually validated at the point of use. This may exceed the policy's required minimum. An input with insufficient evidence is rejected unless the policy explicitly permits a lower-level fallback. An erased envelope projection supplies no issuer proof by itself (L2), but a policy can use it at U if it explicitly accepts unsigned local provenance. A rejected input has no validated level in this model.

Attack model. An attack uses compromise actions, each with a non-negative cost. A direct target is a slot whose check an action attacks, for example through a compromised issuer key or a store edit. The same action can directly target several slots. An attack may also change dependent inputs as a consequence of a direct target, for example by using a key accepted under a substituted mandate. Each outcome-changing attack in this model has a non-empty set of direct targets. The model stipulates a common cost schedule $K : \Lambda \to [0, \infty]$ with $K(\bot) = 0 \leq K(U) < K(E) < K(Q)$. For each direct target t, at least one associated action costs at least $K(\chi(t))$. This is a cost premise about the particular policy, including any available fallback. An assurance label or a successful validation alone does not establish it. **Assumption A2 (substitution):** for each feasible accepted decision input attainable by targeting just slot t, an attack with the same single direct target can produce that input at cost $K(\chi(t))$. The policy and all scope, issuer and validity checks stay fixed. **Assumption A3 (independence):** for distinct direct targets in an attack, the actions that meet the per-target cost floor can be chosen distinct. The attack cost is the sum of its action costs. Define the *attainable assurance* $\alpha(o)$ as the infimum of the costs of attacks that produce a different accepted outcome, with $\inf \emptyset = \infty$. The paper uses the ordering of K in the single-target bounds and its numeric values in additive comparisons.

Authority facts. A policy can require $\text{Auth}(t)$ to verify that a key belongs to a listed service, a register speaks for a company, a signer holds a mandate covering t, or a laboratory is accredited for a test type. These facts are *authority edges*. A policy may require independent authorisations jointly. Even when there is just one authority slot, a replacement grant must satisfy all other policy checks before an attacker can use it to supply a dependent fact.

A set X of slots is *flipping* if some outcome-changing attack has exactly X as its direct targets. Slot t is *singly influential* if $\{t\}$ is flipping. Write Infl for the union of the inclusion-minimal flipping sets. A set of direct targets need not contain every slot whose accepted value changes.

Theorem T3. (a) $\alpha(o) \geq \min_{t \in \text{Infl}} K(\chi(t))$, with $\min \emptyset = \infty$. (a') Under A3, $\alpha(o) \geq \min_{X \text{ flipping}} \sum_{t \in X} K(\chi(t))$. (b) Under A2, $\alpha(o) \leq K(\chi(t))$ for every singly influential slot t. (c) Suppose a feasible attack on one authority slot a can install an accepted grant to an attacker-controlled key and yield a different accepted outcome through dependent evidence, without directly targeting any other slot. Then a is singly influential and, under A2, $\alpha(o) \leq K(\chi(a))$.

(d) Under A2, reaching $\alpha(o) \geq K(Q)$ requires $\chi(t) = Q$ for each singly influential slot. It suffices that $\chi(t) = Q$ for every $t \in \mathrm{Infl}$ under the cost premise of (a).

Proof. (a) Every outcome-changing attack has a non-empty flipping set X. Since the slots are finite, X contains an inclusion-minimal flipping set, so it contains some $t \in \mathrm{Infl}$. The attack pays at least $K(\chi(t))$ for an action associated with that target. Taking the infimum gives (a). If there is no outcome-changing attack, both sides are infinite. (a') Under A3 the actions counted for the targets in X are distinct, so their cost is at least the sum. Minimising over X gives the bound. (b) There is a feasible outcome-changing attack with direct target set $\{t\}$. A2 gives one with the same accepted input at cost $K(\chi(t))$. (c) The new grant and dependent content may both change, but the attack has direct target set $\{a\}$. Part (b) applies. The premise excludes a grant that fails any independent key, issuer, scope or co-authorisation check. (d) If a singly influential slot has level below Q, (b) and strict ordering below Q put the bound below $K(Q)$. If every slot in Infl has level Q, (a) gives the target.

A single compromise can target several slots, as when one issuer uses the same signing key for two reports. The additive bound then needs separate justification. A2 is also a substantive condition. The cheapest corruption of a slot may otherwise produce only immaterial changes. An authority edge supplies the bound in T3(c) only when the policy admits the stated redirection and the changed dependent fact alters the accepted outcome. A derived output can depend on several direct targets. Its assurance follows from that dependence and the attack costs, rather than from a level assigned to the derived value. Figure 5 illustrates the assurance distinction.

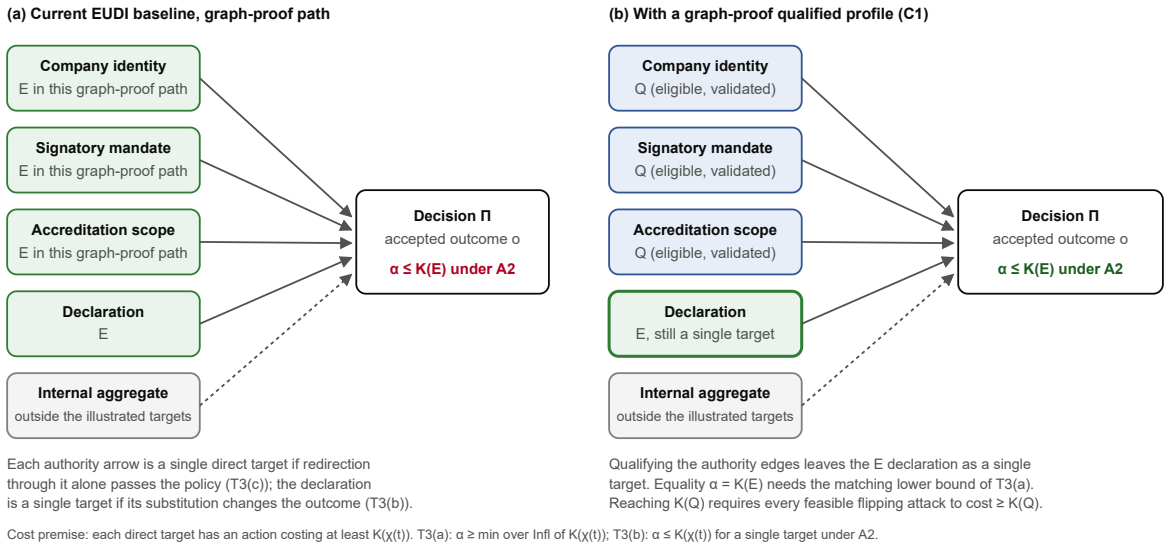


Figure 5: An illustrative policy under A2. (a) An authority edge validated at E bounds the decision by $K(E)$ if it is a single direct attack target satisfying T3(c). (b) If graph-profile issuance and validation make those edges Q, an E declaration that can independently change the accepted outcome still bounds the decision. Equality requires a matching lower bound from T3(a).

Corollary C1 (qualification under the single-pipeline objective). Let the relying party require a homogeneous graph-proof path under Definition 4.1. Let $\mathcal{A}$ be the authority slots meeting the one-target redirection condition of T3(c) for the decision. Assume A2 and the cost premise of T3.

(a) *Required levels.* Reaching $\alpha(o) \geq K(Q)$ requires level Q for each singly influential direct target, including every $a \in \mathcal{A}$. Level Q for all of Infl suffices under the cost premise. For mixed levels, evaluate the feasible attacks. Suppose both reports remain accepted evidence and the

decision returns the Boolean OR of their values. If both values are initially true and substitutions are independent, reversing the outcome requires changing both values to false. If these are the only relevant attacks and their costs attain the slot prices, two E reports yield a cost of $2K(E)$. Raising either report to Q gives a cost of $K(E)+K(Q)$, even though neither report is a singleton target. Audit and other decisions can impose additional requirements.

(b) *Where the edges must sit.* To validate an edge at Q on the required path, the relying party needs an eligible attestation in the path's graph-proof profile, within the model's legal scope (Section 4.2). A source-backed edge checked under a separate envelope profile or a delegate's result lies outside this objective.

(c) *Under the current baseline.* The referenced issuance profiles do not include the graph-domain VCDM profile of A. If no eligible profile is used and the policy accepts an edge $a \in \mathcal{A}$ at E or below on that path, $\alpha(o) \leq K(E)$. A policy requiring Q can instead reject the unavailable evidence. A relying party that permits a mixed path can verify a compliant SD-JWT VC or mdoc with its source proof and a governed mapping (L3(i)). It validates at Q only when the attestation meets the applicable eligibility, issuer, scope, status and recognition checks. The cost depends on which checks it repeats and when.

(d) *Eligibility.* Admitting a graph-proof profile removes the format barrier for the relevant attestations. Qualified issuance and validation also require the conditions in Section 4.2. Once the policy accepts the edges, the decision meets the target if every feasible outcome-changing attack costs at least $K(Q)$. Non-qualified content may remain on the same path when the policy and the cost target permit it.

Proof. (a) The stated necessity and sufficiency claims follow from T3(d). For the independent example, both substitutions are necessary, and additivity and the stated feasibility give the two costs. (b) By Definition 4.1 the edge must use the adopted profile. For that edge to be validated at Q on the adopted path, its profile must be eligible under Section 4.2. L3 describes verification outside that design choice. (c) For A, Section 4.2 supplies the format ceiling. If the policy accepts the edge at E or below and T3(c) applies, the bound follows. Rejection is the other policy case. Source-backed verification can establish the original proof and mapping as in L3(i), and the legal conditions remain separate. (d) Eligibility removes one necessary restriction. The attack-cost condition follows from the definition of α.

C1 separates the choice of verification path from the assurance target. A relying party can use graph-native verification for evidence issued under one graph-proof profile. It can also retain and check qualified source evidence under a mixed-profile policy. In the use cases, authority credentials can be reused across many products. The resulting qualification effort depends on the number of organisations, signatories, scopes and renewals. T2 estimates the verification cost only for its specified workload.

Corollary C2 (evidence across jurisdictions). Let a relying party in J_1 consume evidence issued in J_2. (a) If the relying party cannot validate the source profile or anchor at Q, it rejects evidence requiring that level or applies an authorised lower-level policy. If it accepts a slot at E or below and that slot is singly influential, T3(b) bounds the decision by $K(E)$ under A2 and the cost premise. The same reasoning applies to an accepted authority slot that separately meets T3(c). (b) An intermediary that attests a conversion without a verifiable source binding supplies its own assertion. A projection can preserve source attribution if the relying party checks the retained proof and the authorised mapping. (c) A validated level in J_1 depends on the receiving policy and its checks. For Q, the source attestation must meet the receiving rules for profile, issuer, scope, validity, status and applicable legal recognition. A homogeneous path also requires the agreed verification profile. *Proof.* (a) Section 4.2 gives the policy cases. The applicable singleton bound is T3(b), with T3(c) for an authority redirection. (b) By the derived-fact rule of Section 4.2 and the source-backed check of L3(i). (c) These are the receiving requirements of

Section 4.2 and Definition 4.1. In the EU the legal basis may be Article 14 for third-country services, or Article 24a(9) and Article 45b(3) for the respective recognised attestations within the EU [32].

For a receiving EU path that requires a homogeneous graph-proof profile, that profile must be eligible for the qualified attestations used on the path. Table 1 gives technical examples of foreign evidence. It does not establish legal recognition.

Worked example: web markup. A policy can consume ordinary unsigned schema.org markup at U when it accepts the provenance. If an attacker can change the accepted outcome by editing that markup alone and the model's cost assumptions hold, T3(b) bounds assurance by $K(U)$. A Data Integrity proof by an identified issuer can supply evidence at E if the policy accepts the issuer and validates the proof. Assignment of Q has the additional conditions in Section 4.2.

Table 4: Summary of A and the envelope-only B1 path. The dual-proof nested example in Appendix C retains an inner graph proof and is outside the comparison in this table.

Property	**A: VCDM 2.0 + Data Integrity (RDFC)**	**B1: envelope-only JSON-LD in SD-JWT**	**Result**
Graph semantics	RDF dataset	RDF dataset after expansion	Section 4.1
Verification after merge	graph-native if the full signed inputs are reconstructed	source-backed when the protected input and required openings cannot be recovered from the graph	L2
Meaning bound by the proof	checks the admitted canonical input	signature check insensitive to external context drift, subject to further profile checks	L1
Type systems	RDF classes and application schemas require compatible rules	vct plus RDF classes and application schemas require compatible rules	P1
Disclosure soundness	for verified issuer payload quads under the suite and profile checks	can fail when disclosing a member changes the RDF interpretation	P2
Use-bound verification	if the application consumes the verified view or a specified derivation	if the profile fixes the authenticated reconstruction and its interpretation	P4
Selective disclosure	suite-defined quad groups, with conditional proof unlinkability for BBS	JSON members and array elements, with linkage on reuse of the same issuer JWT	P3, P5
Fully disclosed reuse with refreshed mapping	lower estimated cumulative cost beyond R^* when $\delta > 0$	envelope processing and RDF expansion under M1	T2, Table 3

Continued on next page

Property	A: VCDM 2.0 + Data Integrity (RDFC)	B1: envelope-only JSON-LD in SD-JWT	Result
Cost at issuance and first ingestion	higher for the tested paths	lower for the tested paths	T2, Section 5.5
Source-only envelope check under a trusted mapping	full store proof check	checks the retained source. The timing excludes a check of the cached graph or its source binding	Section 5.5
Selective-disclosure timing	measured derived proof verification	faster verification in the tested implementation under its proof procedure	Section 5.5
Validated level of authority edges under the current baseline	the graph-proof VCDM profile of A is outside the referenced issuance profiles	Q possible with an eligible and validated source attestation, while an erased projection may be rejected	C1(c)
Qualified edges with non-qualified facts	homogeneous when both use the same eligible graph-proof profile and pass the policy	a path mixing A with a separate envelope profile is heterogeneous	L3, C1
Imported evidence	Q requires a profile, authority and recognition accepted in the receiving jurisdiction	the same receiving-policy requirement applies to envelope evidence	C2

6 Use cases

6.1 UC1: supply-chain analytics on digital product passports

Setting. A vehicle manufacturer and its analytics provider assess battery passports and the evidence behind them: carbon footprint, recycled content, raw-material due diligence, supplier identity and the authority of signatories. Article 77 of the Batteries Regulation requires a battery passport from 18 February 2027 for light means of transport batteries, industrial batteries above 2 kWh and electric vehicle batteries placed on the market or put into service. Annex XIII lists carbon-footprint, recycled-content and responsible-sourcing information, with differentiated access. A separate provision sets 18 August 2027 as the start date for the due-diligence duty [58,85]. In the scenario, issuers in several jurisdictions supply evidence for analysis, regulatory reporting and supplier risk scoring. The workload is a projection from the assumptions below.

Decision rule. The analytics provider reports a carbon-footprint performance class in the manufacturer's submission. In this example, its policy checks the manufacturer's identity, the declaration signatory's mandate for that purpose, and a supporting verification statement from a body accredited for the relevant carbon-footprint work. It checks the passport's digest links and escalates if the checks fail. The verification statement attests to the method and its scope. The example assumes that the verification statement does not bind the exact reported class and

that the provider does not independently recompute it. This policy is a scenario assumption. In the restricted-access subcase, the verification body alone receives the supplier's cell-level process data.

Evidence graph (Figure 6). The passport links by digest to a carbon-footprint declaration, a recycled-content declaration, a due-diligence report and supplier credentials. The declaration links to a third-party verification statement, the recycled-content declaration to the recycler's attestation, and the due-diligence report to a smelter audit. Authority edges include the manufacturer's identity, the declaration signatory's mandate, the supplier's identity and the verification body's accreditation. The analytics layer retains each source dataset, with its graph positions and proof information.

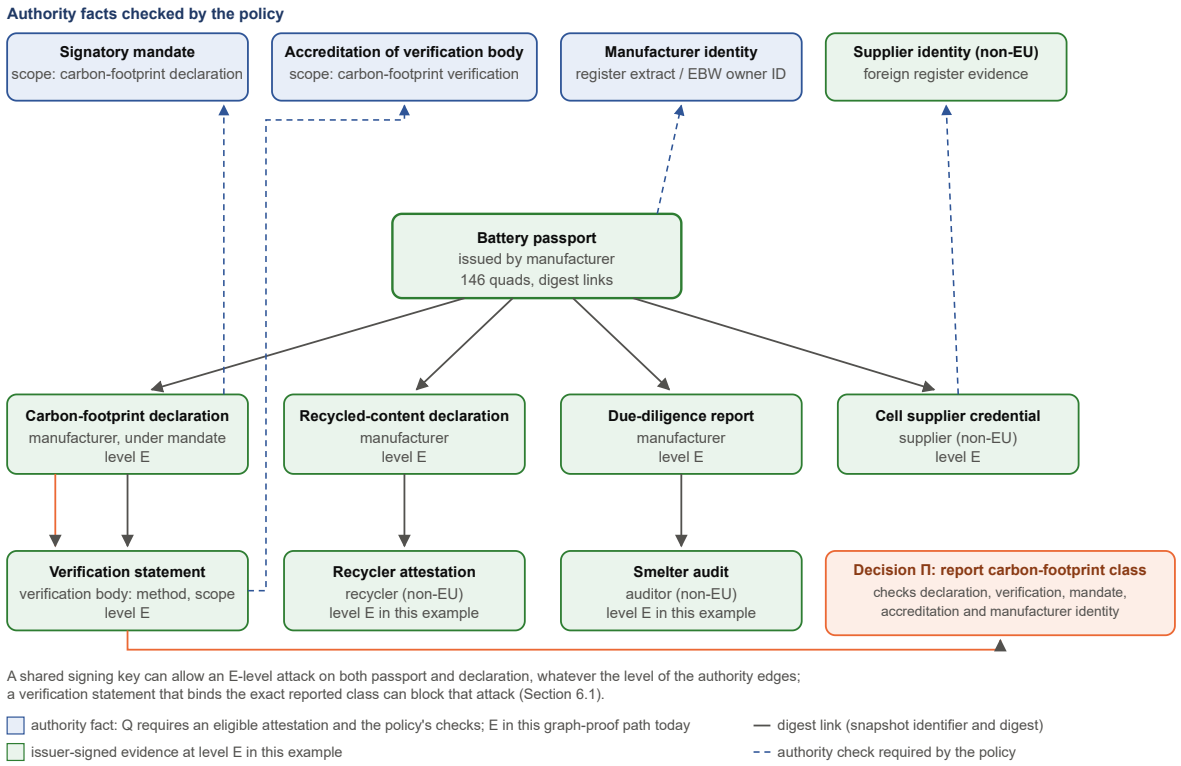


Figure 6: Evidence graph behind one battery passport. Solid grey arrows are digest links. Dashed arrows show authority checks. The orange path shows the declaration and verification statement used by the reporting decision. The example applies the decision rule in Section 6.1.

Volumes (assumptions). 10^6 units in scope for a large manufacturer group, times 10–100 linked credentials per unit, gives 10^7 to 10^8 credentials per year. For a sector-wide service that also ingests traceability events, we assume 10^9. Each credential is modelled with 146 quads and $R = 1, 3, 10$ independent re-verifications, for example by a supplier, manufacturer, analytics provider, authority or auditor. Table 5 assumes the same schedule for A and B.

Table 5: UC1 projected processing time per year, calculated from the 146-quad IRI-only run-3 timings on one thread. The core-hour equivalents include issuance. The per-quad denominator counts the credential's quads on each verification event and excludes issuance. The B^s columns use the same denominator to compare elapsed work, although subsequent B^s checks the envelope and relies on the earlier view.

Credentials per year	R	**A core-hours**	**B core-hours**	**B/A**	B^s **core-hours**	B^s**/A**	**µs per quad-event, A**	**B**	B^s
10^8	1	145	74	0.51	52	0.36	10.0	8.4	5.6
10^8	3	160	133	0.83	66	0.41	6.0	7.8	3.7
10^8	10	215	340	1.58	115	0.54	3.4	7.5	2.4
10^9	10	2,151	3,401	1.58	1,155	0.54	3.4	7.5	2.4

For this workload, B with fresh expansion takes about 17 % less estimated processing time than A at three re-verifications. The calculated crossover is 4.24, so A costs less from five re-verifications onwards. At ten, B takes about 1.6 times A's time. The envelope-only scenario uses 36–54 % of A's time at the values of R shown, conditional on a trusted cached mapping and excluding the cost of retrieving or checking it. The reciprocal batch medians give about 940 credentials (137,000 quads freshly expanded) per second for B with fresh expansion, 3,900 envelope checks per second without expansion, and 3,500 credential checks (0.52 million quads) per second for A. Status checks and network I/O are outside these figures.

Assurance. The policy rejects or escalates if the merged store cannot authenticate an issuer and the source is unavailable. Suppose that a single compromise of the manufacturer's E-level signing key lets an attacker replace the declaration and the passport, including its digest link, with a different reportable class. Assume the same key signs both credentials and that the replacement passes the other checks, including the verification statement's scope. This gives an accepted outcome-changing attack, so $\alpha(o) \leq K(E)$ if that action costs at most $K(E)$. Qualifying the mandate and accreditation alone leaves this attack available. The attack changes more than one credential, so an additive bound under A3 does not apply. If instead the verifier authenticates the exact class and the policy checks agreement, the same attack fails. The attacker would need another feasible route. The example shows why a policy must establish whether an authority or content substitution can change the accepted outcome. Separately, the receiving party must check a foreign recycler's attestation against its format, authority and status requirements. A claim to legal recognition at Q also needs the applicable legal basis (C2).

Qualification volume (assumptions). Suppose the policy calls for qualified attestations of organisation identity, signatory mandates and accreditation where relevant. The workload assumes 10^6 batteries, around 10^3 organisations and several qualified credentials per organisation, with the count depending on signatory, scope and renewal. This gives roughly 10^4 qualified credentials for 10^7 to 10^8 self-asserted credentials. This count does not establish a decision-level assurance target. Where the edges are issued in SD-JWT VC and the content uses graph proofs, the analytics provider can retain the source, verify it and govern the mapping, or rely on delegated validation under its policy (L3). A provider choosing the single-pipeline objective requires an eligible graph-proof profile for any qualified edges on that path (C1).

6.2 UC2: e-commerce customs release

Setting. A consumer in the United States orders children's pyjamas from an overseas seller on an online marketplace. The manufacturer registers the product, the marketplace records the order and invoice, a fulfilment centre packs the parcel, a carrier or customs broker files the waybill and advance entry data, and US Customs and Border Protection (CBP) decides at the port of entry whether to release the parcel, hold it for examination or refer it to a partner government agency, here the Consumer Product Safety Commission (CPSC). CBP reports nearly four million low-value shipments entering the United States each day and names smuggling, duty evasion and counterfeit goods sold through online platforms as the targeting problem [86]. The design follows the package journey demonstrated at the DHS Science and Technology Silicon Valley Innovation Program showcase in March 2021, in which the author's company took part: parties with verified legal-entity identifiers issue credentials for product, order, invoice, packing list, product receipt, package and waybill, and a customs system links the submissions and shows inconsistencies to officers [42, 65]. The credential types align with the W3C Community Group traceability vocabulary, which builds on schema.org and GS1 [87]. The data model below is the paper's reconstruction, not the pilot's implementation.

Data model. The pilot's relational model keys a transaction by a shipment identifier with foreign keys to the products, the manufacturer, the shippers, the distributor, the marketplace and the agencies involved, a list of credential identifiers per workflow step, and separately stored, encrypted references to the buyer and the receiver. Each party has a decentralised identifier, its DID document and a legal-entity credential; each credential has an issuer and a subject. In the evidence graph of Section 4 the same model becomes: one named graph per credential, linked by the shipment identifier and by digest to the credential it continues (L2); one authority edge per submitting party from its credential to its legal-entity attestation, for example a vLEI issued under the GLEIF governance framework [88] or, for an EU party, a company-identity attestation in the European Business Wallet; the certificate of conformity linked by digest to the laboratory report it rests on; and the personal data of buyer and receiver outside the graph, referenced by IRI and digest and released under access control. The agency is both relying party and issuer: its inspection, release and hold credentials enter the same graph.

Scale and automation. Per-shipment graphs accumulate in the agency's data store, where targeting algorithms and AI systems can support shipment release, risk selection and officer review. Shared identifiers, vocabularies and a governed RDF profile preserve the meaning of merged statements; source proofs and authority checks preserve attribution. The threats include impersonation and forged submissions, which attack attribution, and genuinely issued but false classification or conformity statements, which attack content. Authenticated sources and the digest links of Figure 7 support provenance checks and traceability to the responsible issuer. Corroboration, consistency checks and risk-based examination remain necessary to assess the claims themselves. Automation can scale both the benefits of these checks and the consequences of accepting unsupported evidence.

Decision rule. Children's sleepwear in sizes 0 to 6X is subject to the flammability standard of 16 CFR Part 1615, sizes 7 to 14 to Part 1616, with defined exceptions for qualifying infant and tight-fitting garments. Those garments remain subject to the applicable general clothing flammability requirements. For children's products subject to an applicable safety rule, the manufacturer or importer must issue a Children's Product Certificate, supported by CPSC-accepted third-party testing where required, subject to applicable testing exemptions [66, 67]. CPSC implemented certificate eFiling for most regulated imports on 8 July 2026; the corresponding date for products entered from Foreign Trade Zones is 8 January 2027 [89]. In this example the release policy checks: the legal-entity credential and status of every submitting party; that the product credential links by digest to a Children's Product Certificate which links to a test report from a laboratory on the accepted list, or that evidence establishes an applicable exception and compliance with the remaining product requirements; and that GTIN, tariff classification, quantity and package identifier agree across order, invoice, packing list, package and waybill. Any failed or missing check holds the parcel; a failed product check refers it to CPSC. This policy is a scenario assumption.

Evidence graph (Figure 7). Five party credentials form the authority row. The transaction credentials link in workflow order. The certificate links to the laboratory report, and the laboratory's acceptance is a further authority edge. Personal data sits outside the graph.

Cost (projection from Table 3). A shipment carries about eight credentials of around 50 quads. From the run-3 timings for that size, first verification and ingestion of the eight credentials by the customs system costs about 16 ms for A and 7 ms for B on one thread. Three further independent checks, by the broker, by targeting and by the agency, add about 3.6 ms for A with fresh graph checks, 19 ms for B with fresh expansion and 3.2 ms for B with the mapping trusted. At four million shipments a day the totals are about 22, 29 and 11 single-thread hours a day respectively. These processing-only projections do not establish production capacity, peak-load latency or the complete trust-validation cost. The comparison in Section 5.5 concerns whether

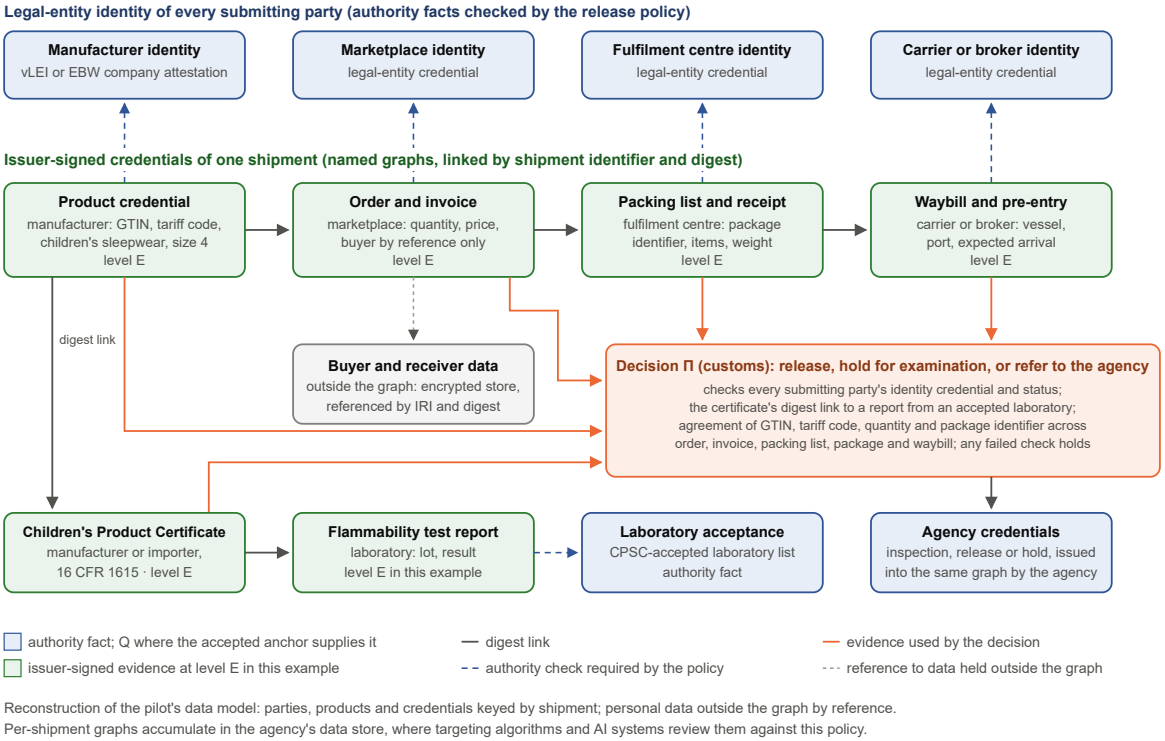


Figure 7: Evidence graph of one e-commerce shipment at the port of entry. Blue boxes are authority facts, green boxes issuer-signed evidence, solid arrows digest links, dashed arrows authority checks, orange arrows the evidence used by the release decision. The data model is reconstructed from the public pilot demonstration.

each check re-establishes the graph from the issuer's proof or trusts a mapping made earlier. Issuance is outside the figures because the parties issue.

Assurance. The release decision relies on self-asserted content: the product's classification and the conformity claim. Under T3 a possible single target is the manufacturer's product credential if the policy accepts its conformity claim at face value; a feasible substitution of the classification or the certificate reference costs at most $K(E)$ and changes duty and agency referral. Linking the certificate by digest to an authenticated report can block that particular substitution only if the policy also checks that the report covers the actual product or lot and the exact conformity claim. A valid digest and accepted laboratory identity alone do not prevent reuse of a report for a different lot. The attack analysis must include this binding and any remaining content substitutions. Identity credentials can also be single targets when their substitution alone changes the accepted outcome, which is why the pilot bound every party to a verified legal-entity identifier. Which anchors the receiving agency accepts for those identifiers, the GLEIF framework, a national trusted list or a future EBW attestation, is the receiving-policy question of C2; the decision needs the identity level, not legal effect. A mis-declared but genuine product remains a content attack outside the authority edges and is the case for examination sampling.

6.3 UC3: an AI safety supervisor in aerospace

Setting. A non-deterministic proposer, an AI agent, proposes engineering or operational actions such as a change to a maintenance task, the acceptance of a design deviation or an operational parameter. A deterministic supervisor permits, rejects or escalates each action against a knowledge graph of applicable requirements: regulation, certification specifications and acceptable means of compliance, industry standards, and the internal requirements of an aircraft manufacturer. This worked design describes no company's system; it follows the research direction of the AERO-CORE project, whose public website was not yet active at the time of writing [90].

The supervisor draws on the Simplex pattern [60] and runtime assurance architectures described in ASTM F3269-21 [91]. EASA's AI Concept Paper Issue 2 gives guidance for Level 1 and 2 machine-learning applications [92]. Earlier proposed Issue 02 lists monitoring an AI/ML constituent and using a traditional backup as one possible mitigation (SRM-02-1) [93]. NASA contractor studies discuss reducing the development assurance required of a complex function by bounding its behaviour at runtime [94]. In its public Halos material, NVIDIA describes monitors that assess proposed driving actions and a robotics design with a safety decision component separate from the main AI compute [95]. Trapp and colleagues developed runtime evaluation of modular safety certificates and dynamic safety management for open adaptive and autonomous systems [96–98]. Dynamic assurance cases connect runtime evidence to the assurance argument [99]. The Guaranteed Safe AI framework uses a verifier to establish a quantitative guarantee that the AI system satisfies a safety specification with respect to a world model [100]. Examples of the knowledge layer include ontology-based compliance checks for airborne software [101] and an OWL-based cockpit assistant [102]. DO-178C and ARP4754B are software and system development-assurance references, respectively [103, 104]. UC3 applies these ideas to requirements authenticated by issuers across organisations.

Decision rule. A proposer asks to extend a component's inspection interval. The supervisor loads the applicable signed base clause and the current, authorised deviation if one exists. It checks scope, version, status and issuer authority, resolves the deviation's digest link to the base clause and compares the proposed interval with the effective limit. With complete, consistent evidence it either retains the existing interval or releases the proposed change to the maintenance approval workflow. A missing, stale, revoked or conflicting input causes escalation or rejection.

Digest links provide a traceability mechanism for this scenario. The supervisor records the result and source versions in a signed audit record. Civil aviation falls under Annex I Section B of the AI Act, where Article 2(2) limits direct application and Article 108 provides the aviation route [59]. The audit record is a design choice.

Two time scales (Figure 8). Local protective functions operate outside the graph-dependent decision path, following a separation proposed in earlier conceptual work on evidence graphs for physical AI [105]. The requirements graph is verified at load and update. Re-verification at decision time depends on policy.

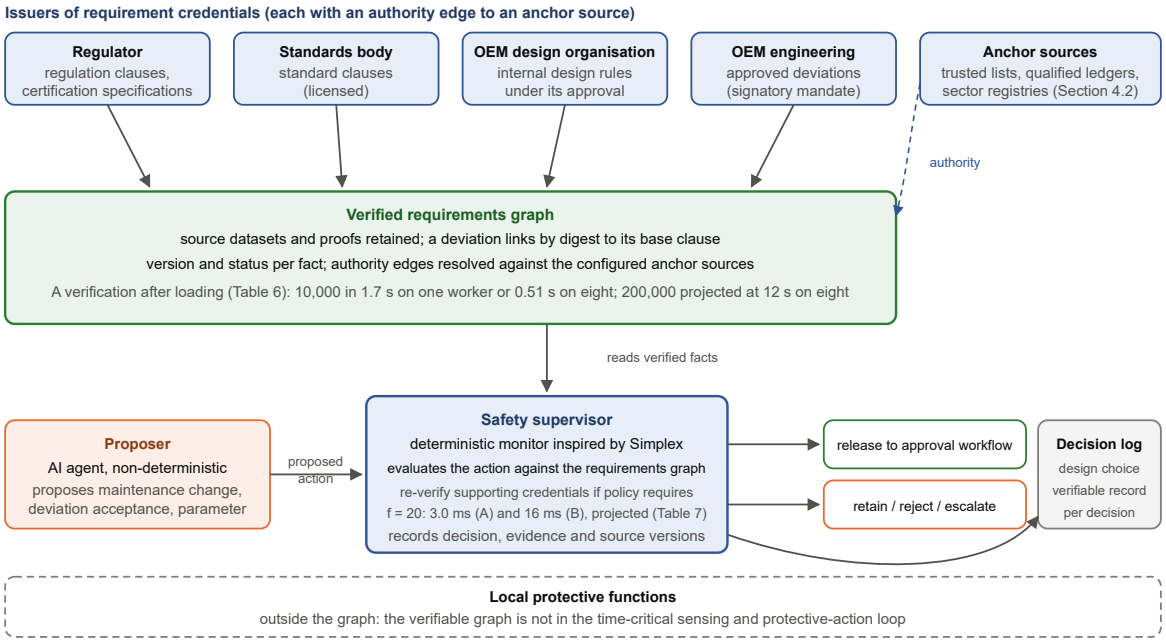


Figure 8: Safety supervisor consuming a verifiable requirements graph. The supervisor evaluates proposed actions against authenticated requirements and records the outcome. Supporting credentials are checked at load and update, and again per decision if the policy requires it. Local protective functions operate separately. Timing examples come from the prototype measurements and the projections in Tables 6 and 7.

Workload and budgets (assumptions). Table 6 gives measured wall time for verifying 10,000 and 50,000 stored requirement credentials with one and eight worker threads, and a linear projection to 200,000. Table 7 gives arithmetic illustrations for decisions with several supporting credentials, using the single-credential observations. The 10 ms and 1,000 ms budgets are design assumptions. The 10 ms example in SOTER is a motion-primitive node period (Section II.B and Fig. 4) [106]. The Black-Box Simplex study reports about 2 ms for reachability under disturbances in a multi-robot example, and separately uses a 2 s control-selection period in its simulated multi-aircraft example (Sections 3.1 and 3.2) [107]. A cockpit-assistant study uses a 1,000 ms reasoning requirement in its Fig. 1 [102]. These examples concern different tasks. In a restricted-access subcase, the supplier allows the manufacturer's supervisor to read a proprietary rule under a confidentiality agreement.

Table 6: UC3 elapsed time to verify every already-ingested 50-quad IRI-only credential once in the prototype, with one timed trial per worker configuration. The clock runs from dispatch until the last worker reports completion and excludes store construction and warm-up. The 200,000 row projects the observed 50,000-credential rates at eight workers, assuming the same elapsed time per credential. Trust checks and network I/O are outside the timed paths.

Requirement credentials	**Workers**	**A from store**	**B via envelopes, mapping refreshed**
10,000	1	1.7 s	9.2 s
10,000	8	0.51 s	2.6 s
50,000	1	8.4 s	50 s
50,000	8	3.1 s	14 s
200,000 (projected)	8	12 s	55 s

The observed eight-worker speed-ups were 2.7–3.7 over one worker across these runs. The projection assumes the larger working set fits in memory at the observed rate. The study did not test that configuration. Single-call prototype ingestion of the 50-quad document into a fresh store cost about 2.0 ms for A. At that rate, 10,000 calls would total about 20 s. Ingesting them into one growing store was outside that timing.

Table 7: UC3 arithmetic projections for f sequential checks of 50-quad IRI-only credentials on one thread. The first value is f times the single-credential median of batch means. Brackets give f times the 99th percentile of the separate single-call sample of 2,000 checks. The bracketed product is a scenario in which each call takes the single-call percentile time. It is neither an aggregate percentile nor a worst-case bound. The budget columns compare only these projections, without policy and status validation, query and decision time or network I/O.

f	**A**	**B, mapping refreshed**	B^s**, envelope only**	**10 ms period**	**1,000 ms budget**
5	0.74 ms [0.89]	4.0 ms [21]	0.67 ms [0.78]	A and B^s estimates fit. For B, only the first estimate fits	all estimates fit
20	3.0 ms [3.6]	16 ms [83]	2.7 ms [3.1]	A and B^s estimates fit. B estimates exceed	all estimates fit
100	15 ms [18]	80 ms [415]	13 ms [16]	all estimates exceed	all estimates fit

Assurance. Suppose a feasible substitution of the current E-level deviation changes the limit so that the supervisor releases a change it would otherwise decline. If the substitute still passes authority, scope, status and digest-link checks and the attack costs at most $K(E)$, T3 bounds the assurance by $K(E)$. The bound still applies if the authority attestations are qualified. A policy that also requires independent approval of the exact deviation can change which substitutions succeed. An authority substitution supplies a separate attack only if the replacement makes the attacker's evidence acceptable under the same policy. When a required proof is unavailable, the supervisor rejects or escalates. The design can record the evidence used and any limiting action established in the analysis. Determining the cheapest feasible attack for a general policy remains separate from producing the audit record.

7 Related work

Signing and canonicalising RDF. Carroll examined RDF graph signing and the difficulty that blank nodes pose [108]. Tummarello, Morbidoni, Puliti and Piazza signed fragments that remain verifiable after merging, including blank-node-connected fragments [109]. Sayers and Karp computed digests of RDF graphs [110]. Kasten, Scherp and Schauß proposed a framework for iteratively signing graph data [111]. Hogan analysed canonical forms for isomorphic and equivalent RDF graphs [112]. Kuhn and Dumontier's trusty URIs reference linked-data artefacts by a hash of canonical content [113], as the graph-profile links in Section 4.2 do. RDFC-1.0 standardises the canonicalisation used by the RDF cryptosuite studied here [6, 23]. We compare the evidence and mappings a verifier needs to retain across the profiled credential formats (Definition 5.1, L2, L3).

Named graphs, provenance and trust. Named graphs can express provenance and support trust policies [114, 115]. PROV-O standardises provenance vocabulary [116], and Artz and Gil surveyed trust models for the Semantic Web [117]. The provenance-preserving merge in Section 4.1 follows the named-graph approach. SPKI certificate theory treats scoped authorisation and delegation [118]. RFC 9334 provides a delegated verification model comparable to L3(ii) [71]. T3's flipping sets are related to minimal cut sets in fault-tree analysis [119] and to work on why-provenance and provenance semirings [120, 121]. We use flipping sets to relate credential acceptance and authority checks to the cost of changing a decision under the specified attacks.

Credential formats and semantic risk. Fujie and Suzuki compare SD-JWT VC, VCDM with eddsa-rdfc-2022 and mdoc in a single issue–present–verify workload. The EdDSA/RDFC subset presentation in their main experiment is re-signed. Their JCS control tests the effect of changing the canonicalisation path, with different proof inputs [84]. Their companion study examines poison graphs, context substitution and loader behaviour in a JSON-LD/RDFC pipeline and distinguishes the tested risks from a general judgement about formats [68]. The studies do not measure verification of stored RDF after a merge or a nested SD-JWT with refreshed RDF mapping. Our prototype times extraction and verification of a stored credential, fresh expansion after envelope verification and a source-only check. The unreviewed working paper *The Semantic Supply Chain* addresses credential meaning across the same three formats [122]. NIST IR 8480 distinguishes JSON-LD credentials following the W3C data model from IETF SD-JWT VC [47].

Runtime assurance and knowledge-based safety. The Simplex architecture, ASTM F3269, Copilot and reinforcement-learning shielding use a separate component to monitor or constrain behaviour [60, 91, 123, 124]. Dynamic assurance cases connect runtime evidence to the assurance argument [99]. Ontology-based compliance checking for airborne software [101] and knowledge-based cockpit assistants [102] use structured knowledge in aviation. UC3 illustrates how a monitor could use requirements authenticated by issuers in several organisations.

8 Discussion

8.1 Expandable, source-backed and graph-native

In a source-backed hybrid, a relying party retains the SD-JWT and relevant disclosures, checks the issuer proof and uses an authenticated interpretation profile for the JSON-LD payload. Claim-oriented consumers can use the profiled claims without expanding RDF. Compatibility with a wallet profile depends on that profile's requirements. A verifier refreshing the graph mapping must establish that the asserted quad follows from the authenticated data under the profile. A verifier trusting an earlier mapping still needs to bind the cached interpretation to

the verified source and apply its policy for freshness and status. The source-only timing in Section 5.5 excludes that binding check.

Fujie and Suzuki distinguish a changed external context that invalidates an RDF proof from one that leaves a JCS or envelope signature over the unchanged reference intact. They argue the latter case from the signing inputs (§VI-A) [68]. L1 states that distinction for the two proof domains. P1 concerns different application type policies, and P2 gives a separate example in which withholding a disclosure changes interpretation. These risks depend on the interpretation profile and the checks performed. C1 applies the single-pipeline objective of Definition 4.1 to a graph-proof path.

Persistent evidence and presentation envelopes. Both architectures can carry JSON-LD. The choice concerns the authenticated proof domain and the retained verification inputs, not JSON syntax alone. A shared VCDM 2.0 profile with RDF-based Data Integrity can align the persistent evidence with graph processing. Shared identifiers and vocabularies, controlled contexts, compatible validation rules, recoverable source boundaries and explicit issuer/status checks remain necessary. If a separately signed inner credential is wrapped only for presentation, policy can verify that wrapper at ingestion and subsequently verify the retained inner source directly. Later decisions relying on the wrapper's audience, holder binding or other assertions must preserve and check the corresponding evidence. Thus nesting does not necessarily impose two proof checks on every subsequent graph read.

8.2 Why qualification matters, and what should be qualified

A QEAA or a PuB-EAA has the same legal effect as a lawfully issued paper attestation within Article 45b(2). Articles 24a(9) and 45b(3) provide for Union-wide recognition of their respective categories. An attestation cannot be denied legal effect or admissibility solely because it is electronic or fails the requirements for a qualified attestation (Art. 45b(1)). Article 45e and Annex VI address verification on request against public authentic sources under the applicable law [32]. Product liability law provides for proportionate disclosure and presumptions in litigation [64]. The AI Act contains dataset governance and logging rules for covered high-risk systems [59]. Machinery rules apply to specified systems using self-evolving machine learning for safety functions (Table 2) [62, 63]. The attestation checks in the worked designs are policy choices informed by those contexts.

Authority edges. A policy can require a qualified attestation for a company's identity, a representation mandate or the scope of a conformity assessment. Under A2, an authority slot can cap decision assurance when changing that slot alone permits an accepted substitute that changes the outcome (T3(c)). Company data and representation powers or mandates are listed in Annex VI. Extending the approach to conformity-assessment certificates, accreditation or agent delegation is a proposal whose scope depends on the applicable authority and scheme [1, 32]. In UC1 the number of authority credentials is a scenario assumption.

Self-asserted facts. Product data, measurements and traceability events can remain assertions of their issuers if the policy accepts their level and the residual attacks meet the target. A feasible direct attack on a content slot, or a low-cost joint attack, can require additional controls. C1 gives conditions for this assessment. The paper's proposal concerns the profile for authority attestations where the policy requires qualified issuance.

Assurance mix. Ordinary EAAs can dominate workloads with repeated product, event and AI-evaluation attestations and reusable authority credentials. UC1 assumes one qualified credential per 1,000–10,000 ordinary credentials. For high-risk use cases, critical infrastructure, regulated industries and sensitive processes, the author supplies a Spherity research estimate of 1 QEAA per 5 ordinary EAAs. This is a use-case estimate, not a measured market ratio or a legal

prescription. The underlying study and methodology are not included in this package. Neither ratio establishes decision assurance; the dependencies and feasible attacks in T3 remain decisive.

Industrial and AI interoperability. GS1 EPCIS supports JSON-LD, and UNTP specifies W3C VCDM 2.0 in compacted JSON-LD with enveloping proofs [11,39]. Their technical adoption does not make every event or graph statement an EAA. We expect AI knowledge graphs to increase the volume and reuse of attestable evidence; no growth rate is measured here. QEAA format standardisation should accommodate shared identifiers, RDF and JSON-LD interoperability, preserved source proofs and mixed assurance levels, so qualified and ordinary evidence can be used together across different workloads.

One pipeline. For the single-pipeline objective in Definition 4.1, the relying party chooses a common graph-proof profile for the path. If a decision requires a qualified authority attestation on that path, a profile eligible for qualified issuance is a prerequisite under the wallet baseline (C1). Common parsing and proof rules can reduce implementation work, but key resolution, status and recognition still require explicit rules. The cost estimates in Section 5.5 compare the chosen verification paths for particular credentials. They do not measure the cost of operating two profile implementations or checking qualified services.

A missing policy-required qualified foundation prevents acceptance of each decision path that depends on it, unless the policy defines an admissible alternative. If every relevant path shares that deficient foundation, the assurance claim for the entire decision graph fails for that purpose. Independent evidence remains verifiable, and an ordinary EAA is sufficient wherever the policy accepts that level. Knowledge graphs express meaning, evidence graphs preserve authenticated sources, and assurance graphs connect claims to requirements and justification. Together they support Trusted AI, trusted industrial AI and trusted physical AI; their presence does not itself establish operational safety or compliance.

In the optional DSCSA credentialing route, OCI specifies embedded proofs for identity and stakeholder credentials and a JWT presentation for the product-identifier verification workflow. Its direct wallet-to-wallet presentation workflow uses a different securing method [48, 49]. This illustrates how a persistent issuer proof can coexist with a proof for presentation.

8.3 What does not change for qualified trust service providers

The proposed profile would retain the applicable eIDAS rules for issuer eligibility, conformity assessment, trusted listing, validation and recognition. It would specify how the graph proof binds the relevant key and service information. A relying party can configure other evidence sources for roles they are authorised to perform. An ecosystem registry may establish membership or scope in its own scheme. A qualified electronic ledger can support integrity and ordering for recorded data under its legal presumption. Neither role, by itself, makes the source an issuer of a qualified attestation [32, 69].

8.4 Presentation and reuse

SD-JWT VC and mdoc support selective disclosure of attributes in wallet presentations. Section 5.5 measures prototype full-disclosure paths and reports a separate presentation comparison. The tested SD-JWT library verifies its presentations faster than the tested Data Integrity library verifies its derivations. The fixtures select the same named subject properties. The presentations have different proof structures and the implementations perform different checks. Data Integrity selective-disclosure suites support deriving presentations over issuer-authenticated quads, subject

to the grouping and privacy conditions in P2 and P3. A profile that uses both envelope and graph proofs needs to specify what each proof covers and how its validation affects reliance (P5).

8.5 Global interoperability

The examples in Table 1 support different conclusions. UNTP 0.7.0 specifies a W3C VCDM 2.0 representation with a JOSE proof. The DHS profile and the optional OCI route illustrate different approaches to W3C credentials. The CBP test and the cited cross-border trade pilots establish activity with verifiable or electronic trade records. The cited sources do not establish a shared credential format across the pilots. A graph-proof option could serve applications that need graph-native verification. Across borders, the relying party must still check the source, mapping, issuer authority and status. Legal recognition has its own requirements (C2).

8.6 Hybrid signatures and post-quantum migration

Post-quantum migration can increase proof size, key-management work and verification steps. NIST FIPS 204 standardises ML-DSA, but selecting a post-quantum algorithm alone does not specify a credential cryptosuite, hybrid construction or eligible qualified issuance profile [125]. A common graph-proof profile could carry traditional and post-quantum components within one credential; the authenticated credential dataset and each proof configuration would still have to remain recoverable.

For a two-component design, the verifier must identify the required algorithms and keys, check their binding to the issuer and the same credential version, reconstruct each component's authenticated input, verify every mandatory component, and apply status, scope and qualified-service checks. An unordered collection of proofs can lose a member without making the remaining signatures invalid. The policy must therefore detect missing required components. Proof chains express dependencies, but neither sets nor chains alone establish a secure hybrid combiner. Where both components are required, accepting either one defeats that policy. RFC 9955 analyses stripping, downgrade and compatibility trade-offs; it is not a credential implementation profile [5, 7].

For nested or mixed credential representations, that work sits alongside envelope decoding, disclosure reconstruction and validation of the source-to-graph mapping. The implementation must define which layer authenticates which content and how algorithm changes affect retained evidence. This increases the combinations that need implementation, testing and audit; it is an engineering inference, not a measured vulnerability rate. Multiple signatures are also available in JWS JSON Serialization and COSE_Sign [126, 127]. Their availability does not automatically extend an SD-JWT VC or mdoc profile. The argument for a shared graph profile is fewer interpretation boundaries, not exclusive access to multiple signatures.

The existing timings use traditional algorithms and cannot quantify PQC overhead or predict a hybrid crossover. A new benchmark must cover complete proof processing, key and certificate material, status checks, storage and transport. Long-lived evidence also needs a migration and preservation policy for retiring algorithms and keys. More signatures cannot turn an ordinary EAA into a QEAA, and format simplicity cannot replace cryptographic validation, qualified issuance or safety engineering.

9 Regulatory and standards path

1. **W3C.** Develop a high-assurance profile for VCDM 2.0. We propose the Recommendation track and leave the choice of a separate deliverable or coordination with future data-model work to the Verifiable Credentials Working Group [128]. The profile would pin contexts, reject undefined terms, specify the quad sets it admits and check A1 directly where the profile relies on A1. It would prescribe input-size and work limits, and a time limit with a means to interrupt work. Library defaults differ and a work budget that grows with input size does not fix elapsed time (Section 5.5) [68]. Cheap checks on claimed issuer, structure and size would precede canonicalisation. Authentication would still require the proof check. The profile would specify cryptosuites, selective disclosure, status, holder binding, X.509 verification, eligible trust anchors, OpenID4VCI and OpenID4VP bindings, and conformance tests. For hybrid signatures, define mandatory components, key and credential bindings, downgrade resistance, migration rules and negative tests for missing or substituted proofs (Section 8.6).
2. **ETSI ESI.** Extend TS 119 472-1 clause 7 [4]. Retain its requirements for signatures or seals and certificate references, and specify the graph-proof suites, context pinning, dataset admission, key binding, status and tests needed to assess conformance. Extend the W3C presentation and request annexes in TS 119 472-2 [34] to cover graph-proof attestations. For the work on EN 319 482-3, propose separating the meaning of attribute definitions from their encodings [129]. After publication of TR 119 060, assess whether linked attestations, access control and logging require further work [36].
3. **European Commission.** Reference the extended clause 7 in the Annex II standards for EUDI Wallet attestation issuance, with any adaptations needed for QEAA and PuB-EAA. Consider a corresponding reference in future EBW rules. Recital 1 of 2026/1731 anticipates wallet support for W3C VCDM profiles once available [2]. The proposed amendment would allow eligible issuers to use the graph-proof profile for authority attestations, such as legal-person identity or mandates, where the applicable attestation scheme permits.
4. **UN/CEFACT and sector bodies.** Evaluate an RDFC-based Data Integrity profile for linked credentials in UNTP. Evaluate migration of the OCI profile to VCDM 2.0 and Data Integrity [39, 49].

10 Limitations

- T1 concerns the representability of finite fact sets and the conditions for composing them. Other representations can satisfy those conditions.
- L2 analyses direct verification under specified proof recovery and storage rules. L3 gives routes for checking graph assertions against evidence issued in another format. Their applicability depends on the source artefact, mapping, trusted verification evidence and the relying party's policy.
- The measured calls use synthetic inputs and one machine. The store is in memory. The prototypes omit production profile and holder-binding checks, trust-chain and status verification, and network or disk retrieval. Standalone cached-lookup timings are not included in the timed proof paths. The source-only and refreshed-mapping calls do not verify a binding to a cached graph. The published-library timings check different code paths and do not bound implementation bias. The presentation measurements cover different proof structures and checks and do not establish an algorithmic ranking. Worker timings cover preloaded partitions at one credential size. The serial budget calculations

are not measurements of decision latency. The canonicalisation stress test uses selected inputs and reports completion and elapsed time rather than conformance or worst-case bounds. Fujie and Suzuki also report sensitivity to runtime and library versions [84].

- T3 concerns the attacks admitted by the model. A2 is needed for upper bounds stated in terms of the cheapest eligible slot cost. A3 is needed to add costs across several targets, which also requires a quantitative cost scale. A common compromise can defeat additivity. The model excludes denial as an attack outcome and does not model probabilities or judicial assessment of evidence. Finding a minimum-cost attack for a general policy is outside the scope of the analysis.
- C1 separates qualification of evidence from assurance of the outcome. The issuance argument is conditional on the single-pipeline objective and the set of eligible profiles. The assurance argument depends on the specified policy and feasible attack costs.
- C2 separates technical interoperability and source attribution from recognition of legal effect.
- Use-case volumes and latency budgets are assumptions. The per-decision signed record and full-record access for the modelled workload are design choices.
- The PQC discussion is a design analysis. No post-quantum or hybrid path was benchmarked, and no hybrid security proof or qualified-profile approval is claimed. The author-provided 1 QEAA per 5 ordinary EAAs estimate has no supporting dataset or methodology in the supplied package.
- Table 1 is a dated selection of public profiles, guidance and pilots. The available sources do not give adoption rates.
- The UC3 design is compared with the cited literature. The ETSI gap analysis TR 119 060 is cited by its public work item.

11 Conclusion

For industrial workflows that combine evidence from many issuers, retain it across organisations and repeatedly verify it as graph data, we recommend a common W3C VCDM 2.0 profile expressed in JSON-LD and secured with RDF-based Data Integrity as the persistent evidence format. This aligns graph processing with the issuer-authenticated proof domain, provided that source snapshots, graph positions, proof configuration and interpretation rules remain recoverable. JSON-LD syntax alone does not provide these properties. A shared profile can reduce the format mappings and verification rules that implementations must govern, test and audit; the paper does not measure that organisational saving.

When an envelope is justified. Add an envelope for a defined presentation, disclosure or interoperability requirement. An envelope-only construction needs available or reliably reconstructible signing inputs and a governed source-to-graph binding. A nested construction retaining an inner graph proof preserves that verification route, while adding outer verification and binding obligations whenever policy relies on both layers. A presentation-only wrapper need not be rechecked for every later use of independently verified inner evidence. These constructions are not inherently insecure. The measured cost advantage for the graph path applies to the stated repeated-verification workload with refreshed mapping; it does not establish universal superiority over presentations or protected cached mappings.

Qualified issuance and decision assurance. To use the common profile for ordinary evidence and the qualified authority attestations required by policy, standards and issuance rules should make

an appropriate graph-proof profile eligible alongside existing profiles. This recommendation is an engineering choice for the specified workload, not a uniqueness theorem for RDF. Eligibility does not qualify an issuer, make an assertion true or establish a decision's safety. The receiving policy must still validate authority, scope, status and source bindings. The attack model explains when content substitutions, authority substitutions and joint attacks limit assurance; qualification alone does not eliminate them.

PQC and multiple signatures. Data Integrity already supports proof sets and chains [5, 6]. Their use for traditional/PQC hybrids requires a validated construction, key and credential bindings, verification of mandatory components and protection against removal or downgrade [7]. A common graph profile can reduce integration variation while supporting migration. Its cryptographic overhead must be measured separately from the present benchmarks; qualified status, operational safety and compliance remain separate requirements.

The prototypes measured here are research artefacts. At the time of writing, a production implementation of the graph-proof design prepared for the European Business Wallet is in field testing with industrial users in Europe; its results are outside this paper.

Declarations

Competing interests. The author is chief executive and founder of Spherity GmbH, which develops software and services for verifiable credentials, digital product passports and organisational identity. The company took part in the DHS Silicon Valley Innovation Program pilot whose public demonstration informs the design in Section 6.2.

Funding. The author's company funded the research. The work received no external funding.

Data and code availability. This revision includes the manuscript, figures and supplementary design notes. Appendix B identifies the separate benchmark code, pinned contexts and run archives needed to reproduce the reported results. Those files are not included in the supplied publication package. The stated repository is https://github.com/spherity/spherity-research; the review on 9 November 2026 did not locate the benchmark archive in its default branch or a published release. A stable archive link and the applicable software and data licences remain to be confirmed before claiming public reproducibility. The archived W3C contexts and RDFC-1.0 test-suite inputs retain their W3C licences. Separately, the new complete worked examples and public verification scripts in `examples/` reproduce the cryptographic checks in Appendix C; they are not the benchmark implementation or run archives.

Licence. This work is licensed under the Creative Commons Attribution 4.0 International licence (CC BY 4.0).

Verification date. The source manuscript records checks on 8 November 2026. This revision rechecked the core issuance baseline, EBW proposal status, cited PQC and multiple-proof standards, and CPSC points on 9 November 2026. The remaining dated references retain their source verification dates.

Use of AI tools. Claude (Anthropic) supported drafting, literature search and benchmark implementation. OpenAI Codex supported manuscript review and source revision. The author is responsible for the content.

Appendix A. Proof remarks

The supplementary file 02-formal-model.md records earlier design notes. The definitions and assumptions used here are those in Sections 4 and 5. For $\delta \neq 0$, the consumer-side crossover in T2 is $R_v^* = (A_0 - B_0)/\delta$ where it lies in the stated workload. In the run-3 calculations, A_0 and B_0 are timed ingestion events. The component expressions are schematic.

Appendix B. Benchmark

The separately referenced benchmark archive is not included in this publication package. The following describes its reported organisation and the inputs needed for reproduction. Run 3 is in benchmark2/. lib.mjs supplies the generator, loader, keys, prototypes, storage operations and timing functions. bench2.mjs runs the benchmark and worker.mjs runs the preloaded worker partitions. probe.mjs checks selected fixtures, including cross-verification of the full-credential RDF proof. model2.mjs and fig4-run3.mjs generate the report and Figure 4 from the archive. The lockfile records the library versions, including jsonld 8.3.3, rdf-canonize 4.0.1, n3 2.13.12, @sd-jwt/core 0.24.3, @owf/crypto 0.7.0, jsonld-signatures 11.6.0, @digitalbazaar/data-integrity 2.5.0, @digitalbazaar/eddsa-rdfc-2022-cryptosuite 1.3.0 and @digitalbazaar/ecdsa-sd-2023-cryptosuite 3.4.1. The successful run used Node.js 24.21.0 with --max-old-space-size=8192.

The archive records file and parsed-document digests for the local contexts. The file SHA-256 of the credentials v2 context is 59955ced6697d61e03f2b2556febe5308ab16842846f 5b586d7f1f7adec92734. Its SHA-256 after parsing and serialising with JSON.stringify is bf6fd611e6773d58f649459a45dbbb81f94e8cfd058a756644cd5a4a882c4050. The generated context has a parsed-document digest. The full-disclosure prototypes use the fixed Ed25519 seed and the prototype SD-JWT uses seeded salts. Reference SD-JWT issuance uses random salts. The two selective-disclosure implementations use separately generated P-256 keys. The benchmark fixtures do not provide production keys or salts.

The explicitly configured prototype canonicalisation calls use maxWorkFactor: 2 in runs 1–3. This setting in rdf-canonize 4.0.1 differs from the per-node call limit varied by Fujie and Suzuki. Their main JavaScript paths use the URDNA2015 option in 3.4.0, which can produce different output from RDFC-1.0 in edge cases [68, 84, 130]. poison.mjs tests direct canonicalisation of selected shapes, generated credential datasets and archived RDFC-1.0 inputs. It records completion or failure at configured factors 1, 2 and 16 and with no work limit, with skips for costly cells. The W3C input sweep checks completion, not equality with expected output or implementation conformance. The outcomes are archived in poison-results.json, poison-results.md and poison-log.txt.

The reported run uses the timings in results-run3.json, which contains the batch means, summaries of single-call samples, reference-library and selective-disclosure timings, and worker wall times. It does not contain the individual latency samples. See bench-log-run3.txt for the log. The archived failed attempt ran out of JavaScript heap. Its log does not isolate a memory cost per quad or the effect of changing the N3.js version.

Once that archive is available, reproduce the calculations by running node model2.mjs and node fig4-run3.mjs from benchmark2/ in a working copy, with the archived run-2 data in benchmark/. The figure script writes its SVG in the working-copy root. To take fresh timings, use the lockfile with npm ci in benchmark2/, run node probe.mjs and run node --max-old-space-size=8192 bench2.mjs. Use node poison.mjs for the separate canonicalisation stress test. Preserve the supplied files before running these commands because the generators and measurements write

output files. The full timing run in the archive is a single observation under its recorded environment. A new run need not reproduce the exact elapsed times.

Runs 1 and 2 are retained in benchmark/. S0 times A re-reading N-Quads text. S1 estimates parsed-quad verification by summing $\kappa+g+v$. S2 models removal of the fitted context-processing constant for both architectures. S3 combines S1 and S2. These are run-2 scenarios. Section 5.5 separates the run-3 comparison with the archived S1 estimates from the within-run comparison with a sum of run-3 components.

Appendix C. Complete signed credentials and verification

The following are complete, executable demonstration credentials, with actual signatures rather than placeholder proof values. They assert that battery 4711 has a recycled-cobalt share of 16 percentage points. The vocabulary is illustrative. The issuer, public keys and contexts are fixed local test inputs; the example does not establish a real recycler's authority, truth of the claim, or QEAA status. Both variants carry exactly the same signed inner credential. [5, 6, 22]

The supplied `examples/` folder contains the original JSON, exact token, pinned contexts, public verification scripts and results. Verification uses 9 November 2026, 12:00 UTC. No private key is needed or included. The local profile requires the stated issuer, credential types, algorithms and unit, the credential disclosure, matching inner and outer issuers, and both proofs in the nested case. It permits the absence of holder binding and credential status; neither a live revocation service nor a wallet protocol is tested.

C.1 The plain VCDM credential

Listing 1 contains the credential itself. The first context is the W3C VCDM 2.0 context; the inline context defines the material terms and an IRI-valued unit. Here `ex:percentagePoint` means percentage points. The `proofValue` is a real Ed25519 signature in multibase encoding. It covers the suite's canonical credential and proof-configuration inputs, not the indentation of this JSON. The complete-document suite used here, `eddsa-rdfc-2022`, does not provide selective disclosure of inner attributes.

Listing 1: Complete VCDM 2.0 credential with its Data Integrity proof

```
{
  "@context": [
    "https://www.w3.org/ns/credentials/v2",
    {
      "@protected": true,
      "ex": "https://example.org/",
      "MaterialCredential": "ex:MaterialCredential",
      "recycledCobaltShare": "ex:recycledCobaltShare",
      "unit": {
        "@id": "ex:unit",
        "@type": "@id"
      }
    }
  ],
  "id": "https://issuer.example/vc/1",
  "type": [
    "VerifiableCredential",
    "MaterialCredential"
  ],
  "issuer": "https://issuer.example",
  "validFrom": "2026-11-09T00:00:00Z",
  "credentialSubject": {
    "id": "https://id.example/battery/4711",
    "recycledCobaltShare": 16,
    "unit": "ex:percentagePoint"
  },
```

```
  "proof": {
    "type": "DataIntegrityProof",
    "created": "2026-11-09T00:00:00Z",
    "verificationMethod": "https://issuer.example#ed25519",
    "cryptosuite": "eddsa-rdfc-2022",
    "proofPurpose": "assertionMethod",
    "proofValue": "zGaCddi1PXdPWMukSZvRoRtpVMszwmuaSmSWgHRAjf3fXEkumMg2vWDejF
   83qnBqbvmDe4B1FtTQ8omH8CcC42MM"
  }
}
```

C.2 Public verification material

The verifier obtains Listing 2 from its pinned local issuer record, not from an unauthenticated claim in the credential. The controller authorises the embedded public Multikey for `assertionMethod`. Listing 3 is the separate P-256 key pinned for the same example issuer's outer envelope. These are public verification keys, not signing secrets. The standard context documents and their SHA-256 hashes are included in `examples/contexts/`; unknown remote document requests are rejected.

Listing 2: Pinned issuer controller and authorised Ed25519 verification method

```
{
  "@context": "https://w3id.org/security/v2",
  "id": "https://issuer.example",
  "assertionMethod": [
    {
      "@context": "https://w3id.org/security/multikey/v1",
      "id": "https://issuer.example#ed25519",
      "type": "Multikey",
      "controller": "https://issuer.example",
      "publicKeyMultibase": "z6MkjNfty9qLyRo9AgQh71j9QX4H8hYQCyUF8r2Z4W7tvfau
    "
    }
  ]
}
```

Listing 3: Pinned public key for the outer ES256 signature

```
{
  "kty": "EC",
  "x": "98iFb5IBJKje3iOVduWmGWSDXXoW6L4t9f2NqNw9VR0",
  "y": "fbonvgL9uaDKmK_ZlswV1GqqnLf5Xup-bw_Q1ZlSkjU",
  "crv": "P-256",
  "kid": "https://issuer.example#es256",
  "alg": "ES256",
  "use": "sig"
}
```

C.3 The complete nested credential

The outer format uses the `dc+sd-jwt` type and an example-specific `vct` profile. Listings 4 and 5 show its decoded protected header and signed payload. The single `_sd` digest commits to a disclosure containing a random salt, the claim name `credential`, and the entire signed object in Listing 1. It therefore reveals the inner credential as one unit; it does not selectively reveal individual battery attributes. [28, 29]

Listing 4: Decoded protected header of the outer SD-JWT

```
{
  "alg": "ES256",
  "typ": "dc+sd-jwt",
  "kid": "https://issuer.example#es256"
}
```

Listing 5: Decoded signed payload, before disclosure reconstruction

```
{
  "iss": "https://issuer.example",
  "iat": 1794182400,
  "vct": "https://issuer.example/type/nested-material",
  "_sd_alg": "sha-256",
  "_sd": [
    "TqMFz3Asrf4bO_jEX_avXVLNzLfZ-P-KSxU4FnAxcM0"
  ]
}
```

Listing 6 is the complete wire value: issuer-signed JWT, a tilde, the base64url disclosure, and a final tilde. It has no ellipses or omitted signature bytes. Long-value wrapping in these listings is typographical; use the supplied files for exact inputs, including `examples/nested.sd-jwt`. The decoded disclosure and fully reconstructed claims are also supplied as JSON files. The final empty segment means there is no key-binding JWT under this example profile.

Listing 6: Complete nested SD-JWT serialization, including the disclosure

```
eyJhbGciOiJFUzI1NiIsInR5cCI6ImRjK3NkLWp3dCIsImtpZCI6Imh0dHBzOi8vaXNzdWVyLmV4Y
   W1wbGUjZXMyNTYifQ.eyJpc3MiOiJodHRwczovL2lzc3Vlci5leGFtcGxlIiwiaWF0IjoxNzk
   0MTgyNDAwLCJ2Y3QiOiJodHRwczovL2lzc3Vlci5leGFtcGxlL3R5cGUvbmVzdGVkLW1hdGVy
   aWFsIiwiX3NkX2FsZyI6InNoYS0yNTYiLCJfc2QiOlsiVHFNRnozQXNyZjRiT19qRVhfYXZYV
   kxOekxmWi1QLUtTeFU0Rm5BeGNNMCJdfQ.esVwHXL1gvrXOOLrYibjIORj-ZQbuVJXHvr2Iwc
   l9gilcEUfpyV_Lgbo9uyb4eRVlErBO2a6ke0Z0PN6O4WCKA~WyJ2LU5YUzFnQ19YTGZ5SFJNV
   UpfQnpRIiwiY3JlZGVudGlhbCIseyJAY29udGV4dCI6WyJodHRwczovL3d3dy53My5vcmcvbn
   MvY3JlZGVudGlhbHMvdjIiLHsiQHByb3RlY3RlZCI6dHJ1ZSwiZXgiOiJodHRwczovL2V4YW1
   wbGUub3JnLyIsIk1hdGVyaWFsQ3JlZGVudGlhbCI6ImV4Ok1hdGVyaWFsQ3JlZGVudGlhbCIs
   InJlY3ljbGVkQ29iYWx0U2hhcmUiOiJleDpyZWN5Y2xlZENvYmFsdFNoYXJlIiwidW5pdCI6e
   yJAaWQiOiJleDp1bml0IiwiQHR5cGUiOiJAaWQifX1dLCJpZCI6Imh0dHBzOi8vaXNzdWVyLm
   V4YW1wbGUvdmMvMSIsInR5cGUiOlsiVmVyaWZpYWJsZUNyZWRlbnRpYWwiLCJNYXRlcmlhbEN
   yZWRlbnRpYWwiXSwiaXNzdWVyIjoiaHR0cHM6Ly9pc3N1ZXIuZXhhbXBsZSIsInZhbGlkRnJv
   bSI6IjIwMjYtMTEtMDlUMDA6MDA6MDBaIiwiY3JlZGVudGlhbFN1YmplY3QiOnsiaWQiOiJod
   HRwczovL2lkLmV4YW1wbGUvYmF0dGVyeS80NzExIiwicmVjeWNsZWRDb2JhbHRaGFyZSI6MT
   YsInVuaXQiOiJleDpwZXJjZW50YWdlUG9pbnQifSwicHJvb2YiOnsidHlwZSI6IkRhdGFJbnR
   lZ3JpdHlQcm9vZiIsImNyZWF0ZWQiOiIyMDI2LTExLTA5VDAwOjAwOjAwWiIsInZlcmlmaWNh
```

```
dGlvbk1ldGhvZCI6Imh0dHBzOi8vaXNzdWVyLmV4YW1wbGUjZWQyNTUxOSIsImNyeXB0b3N1a
XRlIjoiZWRkc2EtcmRmYy0yMDIyIiwicHJvb2ZQdXJwb3NlIjoiYXNzZXJ0aW9uTWV0aG9kIi
wicHJvb2ZWYWx1ZSI6InpHYUNkZGkxUFhkUFdNdWtTWnZSb1J0cFZNc3p3bXVhU21TV2dIUkF
qZjNmWEVrdW1NZzJ2V0RlakY4M3FuQnFidm1EZTRCMUZ0VFE4b21IOENjQzQyTU0ifX1d~
```

C.4 What the verifier actually does

Plain credential. The machine follows these steps [5, 6]:

1. Parse the JSON and enforce the local profile: expected VCDM context and types, issuer, proof suite and purpose, subject fields, value range, unit and validity time. Load only the pinned contexts and issuer record. Check that the verification method is authorised for the issuer's `assertionMethod`.
2. Separate the unsigned credential from its proof. Prepare the proof configuration without `proofValue`, using the document context. Expand and canonicalise the credential and the proof configuration with the suite's RDF Dataset Canonicalization procedure. Reject processing that would lose undefined claims.
3. Hash both canonical byte strings with SHA-256. Concatenate the proof-configuration hash first and the credential hash second. Decode the multibase signature and public key and verify the Ed25519 signature over that concatenation.
4. Only after the required checks pass, expose the verified claim to the application. Preserve the source credential or its losslessly recoverable dataset, proof configuration and source identity if it will be checked again from a graph store. A merged set of triples with those boundaries erased is not, by itself, sufficient verification evidence.

Nested credential. The machine must first perform the following outer steps [28]:

1. Split the combined serialization, retaining the original JWT and disclosure bytes. Verify the ES256 signature with the pinned outer public key. Enforce the allowed algorithm, protected `typ` and key identifier, issuer, `vct`, issuance time and disclosure-hash algorithm.
2. Hash the original base64url disclosure string with SHA-256 and match its digest to the signed commitment. Do not hash a decoded and reserialised substitute. Reject uncommitted or duplicate disclosures and names that collide with clear claims.
3. Decode the authenticated disclosure and reconstruct its `credential` member. Enforce the profile's required disclosure and inner/outer issuer binding. Decoding alone authenticates nothing; a valid outer signature does not establish the inner proof's validity.
4. Apply every plain-credential step above to the recovered signed credential. Both layers must pass before this profile accepts it. If it is stored as evidence, retain the original envelope/disclosure and their binding to the verified inner source whenever later policy relies on the outer assertion.

C.5 Extra work, tested outcomes and limits

For this one-credential, one-disclosure example, the comparison is concrete. Both paths verify the same inner proof. The nested path adds an outer signature check, a disclosure hash/match, reconstruction and binding checks. The counts below describe these artifacts; they are not elapsed-time measurements.

Work or representation	Plain credential	Nested credential
Signature verification	One Ed25519	One ES256 plus one Ed25519
RDF canonicalisation	Two suite inputs	Same two inner inputs
Explicit SHA-256 work	Two inner hashes	Two inner hashes plus one disclosure hash
Disclosure reconstruction	None	One whole-credential disclosure
Serialized bytes	797 compact JSON	1534 combined SD-JWT

The byte counts use minified UTF-8 JSON for the plain credential and the exact ASCII combined token, excluding external key and context documents in both cases. The explicit hash count excludes hashing internal to signature algorithms and canonicalisation. The whole-credential disclosure also means that removing an inner attribute generally breaks this Data Integrity proof. A different selective-disclosure suite or an envelope-only design would require a different analysis.

The public test runner verifies both complete examples and rejects changed claims, changed proof values, changed context meanings, loss of an unmapped term, removal of assertion-method authority, altered outer payloads or signatures, changed/uncommitted/duplicate/colliding disclosures, and withholding the required credential. A separately signed negative fixture has a valid outer signature and disclosure commitment but an invalid inner proof: outer-only verification passes and the required combined verification rejects it. This directly demonstrates why unnesting and verifying are different operations.

A second check verifies both signatures directly with Node/OpenSSL, reconstructs the signing input and checks the disclosure commitment; it shares the RDF processing dependency and is therefore an independent signature-path check, not an independent RDF implementation. The public scripts and test reports are in `examples/`. Run `npm ci -ignore-scripts` there once, then `node verify.mjs` and `node audit-spec.mjs`; verification uses local pinned inputs.

These additional stages create integration and testing obligations, not a proof that nesting is inherently insecure. In particular, this two-proof example differs from Architecture B1's envelope-only JSON-LD payload, which has no inner Data Integrity proof to verify. If an outer wrapper serves only a presentation step, policy can check it at ingestion and later verify the retained inner credential directly; the extra outer work is not necessarily repeated on every graph read. Reliance on outer assertions still requires their evidence and binding. The earlier benchmark timings cannot be assigned to this new example. Cryptographic success also does not establish legal qualification, production interoperability, truth of the assertion or AI safety.

References

[1] European Commission, Proposal for a Regulation on European Business Wallets, COM(2025) 838. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52025PC0838

[2] Commission Implementing Regulation (EU) 2026/1731 of 15 July 2026 as regards applicable standards and specifications, including Annexes IV and XII. https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32026R1731

[3] Commission Implementing Regulation (EU) 2026/1735 of 15 July 2026 amending Implementing Regulation (EU) 2025/1569. https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32026R1735

[4] ETSI TS 119 472-1 V1.2.1 (2026-02). https://www.etsi.org/deliver/etsi_ts/119400_119499/11947201/01.02.01_60/ts_11947201v010201p.pdf

[5] W3C, Verifiable Credential Data Integrity 1.0, Recommendation, 15 May 2025. https://www.w3.org/TR/vc-data-integrity/

[6] W3C, Data Integrity EdDSA Cryptosuites v1.0, Recommendation, 15 May 2025. https://www.w3.org/TR/2025/REC-vc-di-eddsa-20250515/

[7] N. Bindel, B. Hale, D. Connolly and F. Driscoll, Hybrid Signature Spectrums, IETF RFC 9955, July 2026 (Informational). https://www.rfc-editor.org/rfc/rfc9955.html

[8] W3C, RDF 1.1 Concepts and Abstract Syntax, 2014. https://www.w3.org/TR/rdf11-concepts/

[9] W3C, RDF 1.1 Semantics, 2014. https://www.w3.org/TR/rdf11-mt/

[10] IDTA, Specification of the Asset Administration Shell, Part 1: Metamodel (IDTA-01001), with XML, JSON and RDF mappings. https://industrialdigitaltwin.org/en/content-hub/aasspecifications/specification-of-the-asset-administration-shell-part-1-metamodel-idta-number-01001

[11] GS1, EPCIS and CBV 2.0, 2022. https://ref.gs1.org/standards/epcis/

[12] W3C, JSON-LD 1.1, Recommendation, 16 July 2020. https://www.w3.org/TR/json-ld11/

[13] schema.org, About. https://schema.org/docs/about.html

[14] Schema.org, Yandex now supports schema.org markup, 4 November 2011. https://blog.schema.org/2011/11/04/yandex-now-supports-schema-org-markup/

[15] Google Search Central, Introduction to structured data markup. https://developers.google.com/search/docs/appearance/structured-data/intro-structured-data

[16] HTTP Archive, Web Almanac 2025, SEO chapter, Figure 5.50 (home pages), 15 January 2026. https://almanac.httparchive.org/en/2025/seo#home-pages-structured-data

[17] Web Data Commons, structured-data extraction from the October 2024 Common Crawl, overall statistics table, released 10 January 2025. https://webdatacommons.org/structureddata/2024-12/stats/stats.html

[18] Google, A reintroduction to our Knowledge Graph and knowledge panels, 20 May 2020. https://blog.google/products/search/about-knowledge-graph-and-knowledge-panels/

[19] Google, Expanding AI Overviews and introducing AI Mode, 5 March 2025. https://blog.google/products/search/ai-mode-search/

[20] Google Search Central, AI features and your website. https://developers.google.com/search/docs/appearance/ai-features

[21] Wikidata, Statistics, accessed November 2026. https://www.wikidata.org/wiki/Wikidata:Statistics

[22] W3C, Verifiable Credentials Data Model v2.0, Recommendation, 15 May 2025. https://www.w3.org/TR/vc-data-model-2.0/

[23] W3C, RDF Dataset Canonicalization (RDFC-1.0), Recommendation, 21 May 2024. https://www.w3.org/TR/rdf-canon/

[24] W3C, Securing Verifiable Credentials using JOSE and COSE, Recommendation, version of 13 October 2026. https://www.w3.org/TR/2026/REC-vc-jose-cose-20261013/

[25] W3C, Decentralized Identifiers (DIDs) v1.0, Recommendation, 19 July 2022. https://www.w3.org/TR/did-core/

[26] W3C, Data Integrity ECDSA Cryptosuites v1.0, Recommendation, 15 May 2025. https://www.w3.org/TR/vc-di-ecdsa/

[27] W3C, Data Integrity BBS Cryptosuites v1.0, Candidate Recommendation Draft, 10 September 2026. https://www.w3.org/TR/2026/CRD-vc-di-bbs-20260910/

[28] IETF RFC 9901, Selective Disclosure for JSON Web Tokens, November 2025. https://www.rfc-editor.org/rfc/rfc9901

[29] IETF OAuth Working Group, SD-JWT-based Verifiable Digital Credentials (SD-JWT VC), draft-ietf-oauth-sd-jwt-vc-26, 6 November 2026 (Internet-Draft). https://datatracker.ietf.org/doc/draft-ietf-oauth-sd-jwt-vc/26/

[30] ISO/IEC 18013-5:2021, Mobile driving licence (mDL) application.

[31] OpenID Foundation, OpenID4VC High Assurance Interoperability Profile 1.0, final, 24 December 2025. https://openid.net/specs/openid4vc-high-assurance-interoperability-profile-1_0-final.html

[32] Regulation (EU) No 910/2014 as amended by Regulation (EU) 2024/1183. https://eur-lex.europa.eu/eli/reg/2024/1183/oj

[33] Commission Implementing Regulation (EU) 2025/1569, consolidated text of 11 August 2026, Arts. 3–4 and Annex II. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:02025R1569-20260811

[34] ETSI TS 119 472-2 V1.2.1, Profiles for Electronic Attestation of Attributes, Part 2: Profiles for EAA/PID Presentations to Relying Party, March 2026. https://www.etsi.org/deliver/etsi_ts/119400_119499/11947202/01.02.01_60/ts_11947202v010201p.pdf

[35] European Parliament Legislative Observatory, Establishment of European Business Wallets, procedure 2025/0358(COD), status 7 November 2026. https://oeil.europarl.europa.eu/oeil/en/procedure-file?reference=2025%2F0358%28COD%29

[36] ETSI work item DTR/ESI-0019060, TR 119 060. https://portal.etsi.org/webapp/WorkProgram/Report_WorkItem.asp?WKI_ID=78177

[37] Commission Implementing Regulation (EU) 2026/1778, https://eur-lex.europa.eu/eli/reg_impl/2026/1778/oj/eng ; European Commission, DPP resources and documentation, https://single-market-economy.ec.europa.eu/single-market/digital-product-passport/resources-documentation_en

[38] European Commission, The Digital Product Passport Registry is now live, 20 July 2026. https://single-market-economy.ec.europa.eu/news/digital-product-passport-registry-now-live-2026-07-20_en

[39] UN/CEFACT, UN Transparency Protocol 0.7.0, Verifiable Credentials profile. https://untp.unece.org/docs/0.7.0/specification/VerifiableCredentials/

[40] UNECE, Recommendation No. 49, adopted July 2025. https://unece.org/sustainable-development/press/new-un-policy-recommendation-foster-transparency-and-accountability

[41] DHS, Implementation Profile of W3C VCs and DIDs (work in progress), 29 September 2022. https://lists.w3.org/Archives/Public/public-credentials/2022Sep/att-0253/DHS.W3C.VC-DID.Implemenation.Profile-20220929-SHARE.pdf

[42] DHS S&T SVIP, Plug-Fest report, W3C CCG list, 4 March 2021. https://lists.w3.org/Archives/Public/public-credentials/2021Mar/0022.html

[43] CBP, Global Interoperability Standards test review, May 2024. https://cbp.gov/sites/default/files/2024-05/360%20review%20Summary%20GIS%20Part%201%20%281%29_508.pdf

[44] U.S. Customs and Border Protection, Commercial Customs Operations Advisory Committee, Next Generation Facilitation Subcommittee, ACE 2.0 Working Group, issue paper, September 2024. https://www.cbp.gov/sites/default/files/2024-11/ngf_ace_2.0_issue_paper_-_sept_2024.pdf

[45] Federal Register, CBP Pipeline Global Interoperability Standards Test, 2 June 2026. https://www.federalregister.gov/d/2026-10979

[46] NIST SP 800-63C-4, July 2025, §3.12.1. https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63C-4.pdf

[47] NIST IR 8480 (initial public draft), October 2024, §7.3.3. https://nvlpubs.nist.gov/nistpubs/ir/2024/NIST.IR.8480.ipd.pdf

[48] Partnership for DSCSA Governance, Foundational Blueprint for 2023 Interoperability, Chapter 6: DSCSA Credentialing and User Authentication Functional Design, v1.6, 15 July 2026. https://dscsagovernance.org/wp-content/uploads/2026/07/PDG-Blueprint-Chapter-6-v1.6.pdf

[49] Open Credentialing Initiative, Digital Wallet Conformance Criteria v3.4.1, 12 August 2024. https://open-credentialing-initiative.github.io/Digital-Wallet-Conformance-Criteria/latest/

[50] OpenID Foundation, Reflections on the California mDL hackathons, 16 December 2024. https://openid.net/driving-innovation-and-adoption-reflections-california-mdl-hackathons/

[51] South China Morning Post, Blockchain-managed IDs arrive in China with new government-backed scheme aiming to reduce data leaks, 14 December 2023 (secondary reporting). https://www.scmp.com/tech/policy/article/3245028/blockchain-managed-ids-arrive-china-with-new-government-backed-scheme-aiming-reduce-data-leaks

[52] W3C DID Working Group participants. https://www.w3.org/groups/wg/did/participants

[53] ITU-T Recommendation Y.3087, Self-controlled identity based on blockchain – Functional requirements and architecture, April 2025. https://www.itu.int/rec/T-REC-Y.3087/en. Supporting members are listed in the work programme at https://www.itu.int/ITU-T/workprog/wp_item.aspx?isn=21507

[54] Infocomm Media Development Authority, IMDA's TradeTrust is transforming global trade, China–Singapore pilot example. https://www.imda.gov.sg/resources/blog/blog-articles/2024/01/tradetrust-transforms-cross-border-trade

[55] Singapore Ministry of Digital Development and Information, Singapore and China deepen exchange in digital economy at 2nd Singapore–China Digital Policy Dialogue in Chongqing, 2025. https://www.mddi.gov.sg/newsroom/singapore-and-china-deepen-exchange-in-digital-economy-at-2nd-singapore-china-digital-policy-dialogue--dpd--in-chongqing--china/

[56] Catena-X, CX-0149 Wallet Requirements v2.1.0. https://catenax-ev.github.io/docs/next/standards/CX-0149-WalletRequirements

[57] Regulation (EU) 2024/1781. https://eur-lex.europa.eu/eli/reg/2024/1781/oj/eng

[58] Regulation (EU) 2023/1542 (Batteries). https://eur-lex.europa.eu/eli/reg/2023/1542/oj/eng

[59] Regulation (EU) 2024/1689 (Artificial Intelligence Act), consolidated text of 27 July 2026 incorporating Regulation (EU) 2026/1744. https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:02024R1689-20260727

[60] L. Sha, Using simplicity to control complexity, IEEE Software 18(4), 2001, 20–28. https://doi.org/10.1109/MS.2001.936213

[61] Regulation (EU) 2024/2847 (Cyber Resilience Act). https://eur-lex.europa.eu/eli/reg/2024/2847/oj

[62] Regulation (EU) 2023/1230 (Machinery). https://eur-lex.europa.eu/eli/reg/2023/1230/oj

[63] Corrigendum to Regulation (EU) 2023/1230, including the correction to Article 54. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32023R1230R(01)

[64] Directive (EU) 2024/2853 (Product Liability). https://eur-lex.europa.eu/eli/dir/2024/2853/oj

[65] Spherity, DHS SVIP Plugfest Demo: E-commerce Digital Twin, DID/VC Tech Demo and Linked Data Traceability Chains, and E-commerce Digital Twin for Customs Process Automation, video recordings of the DHS S&T SVIP Interoperability Showcase, March 2021. https://vimeo.com/540163632 ; https://vimeo.com/524276878

[66] U.S. Consumer Product Safety Commission, Flammable Fabrics Act, sections on children's sleepwear and infant and tight-fitting garments, accessed 9 November 2026. https://www.cpsc.gov/Business--Manufacturing/Business-Education/Business-Guidance/Flammable-Fabrics-Act

[67] U.S. Consumer Product Safety Commission, Third Party Testing Guidance, accessed 8 November 2026. https://www.cpsc.gov/Business--Manufacturing/Testing-Certification/Third-Party-Testing

[68] N. Fujie and S. Suzuki, A Security Analysis of Verifiable Credential Verification Pipelines: Attack Surface Arising from JSON-LD/RDFC Canonicalization Compared with SD-JWT VC and mdoc, arXiv:2611.04461v1, 3 November 2026 (preprint). https://arxiv.org/abs/2611.04461v1

[69] European Blockchain Services Infrastructure, Issuers Registry API documentation. https://hub.ebsi.eu/api/core/issuers-registry

[70] IETF RFC 8259, JSON, 2017.

[71] H. Birkholz, D. Thaler, M. Richardson, N. Smith, W. Pan, Remote ATtestation procedureS (RATS) Architecture, IETF RFC 9334, January 2023. https://www.rfc-editor.org/rfc/rfc9334

[72] W3C, Shapes Constraint Language (SHACL), Recommendation, 20 July 2017. https://www.w3.org/TR/2017/REC-shacl-20170720/

[73] B. Schneier, A Plea for Simplicity, 19 November 1999. https://www.schneier.com/essays/archives/1999/11/a_plea_for_simplicit.html

[74] C. A. R. Hoare, The Emperor's Old Clothes, Communications of the ACM 24(2), 1981, 75–83. https://doi.org/10.1145/358549.358561

[75] L. Sassaman, M. L. Patterson, S. Bratus, A. Shubina, The Halting Problems of Network Stack Insecurity, USENIX ;login: 36(6), 2011.

[76] J. Miller, An Exploration of JSON Interoperability Vulnerabilities, Bishop Fox, 2021. https://bishopfox.com/blog/json-interoperability-vulnerabilities

[77] T. McLean, Critical vulnerabilities in JSON Web Token libraries, 31 March 2015. https://auth0.com/blog/critical-vulnerabilities-in-json-web-token-libraries/

[78] S. Arciszewski, "No Way, JOSE!", Paragon Initiative, 14 March 2017. https://paragonie.com/blog/2017/03/jwt-json-web-tokens-is-bad-standard-that-everyone-should-avoid

[79] Y. Sheffer, D. Hardt, M. Jones, JSON Web Token Best Current Practices, IETF RFC 8725, February 2020. https://www.rfc-editor.org/rfc/rfc8725

[80] T. Looker, Multiple significant security vulnerabilities in the design of data integrity, W3C Verifiable Credential Data Integrity repository, issue 272, opened 21 June 2024. https://github.com/w3c/vc-data-integrity/issues/272

[81] R. Verborgh et al., N3.js, npm package n3 2.13.12. https://www.npmjs.com/package/n3/v/2.13.12

[82] Digital Bazaar, jsonld-signatures 11.6.0, @digitalbazaar/data-integrity 2.5.0, @digitalbazaar/eddsa-rdfc-2022-cryptosuite 1.3.0 and @digitalbazaar/ecdsa-sd-2023-cryptosuite 3.4.1, npm packages. https://www.npmjs.com/package/jsonld-signatures/v/11.6.0, https://www.npmjs.com/package/@digitalbazaar/data-integrity/v/2.5.0, https://www.npmjs.com/package/@digitalbazaar/eddsa-rdfc-2022-cryptosuite/v/1.3.0, https://www.npmjs.com/package/@digitalbazaar/ecdsa-sd-2023-cryptosuite/v/3.4.1

[83] OpenWallet Foundation, SD-JWT JS and OWF Crypto, npm packages @sd-jwt/core 0.24.3 and @owf/crypto 0.7.0. https://www.npmjs.com/package/@sd-jwt/core/v/0.24.3, https://www.npmjs.com/package/@owf/crypto/v/0.7.0

[84] N. Fujie and S. Suzuki, An Empirical Evaluation of Signing and Verification Performance of Verifiable Credential Formats: A Cross-Language Benchmark of SD-JWT VC, W3C VCDM, and mdoc, arXiv:2611.03493v1, 2 November 2026 (preprint). https://arxiv.org/abs/2611.03493v1

[85] Regulation (EU) 2025/1561 of 18 July 2025 amending Regulation (EU) 2023/1542 as regards obligations of economic operators concerning battery due diligence policies, Art. 1. https://eur-lex.europa.eu/eli/reg/2025/1561/oj

[86] U.S. Customs and Border Protection, E-Commerce, web page last modified 6 November 2026, accessed 8 November 2026. https://www.cbp.gov/trade/basic-import-export/e-commerce

[87] W3C Credentials Community Group, Traceability Vocabulary v1.0, accessed 8 November 2026. https://w3c-ccg.github.io/traceability-vocab/

[88] GLEIF, vLEI Ecosystem Governance Framework, version 4.0, accessed 8 November 2026. https://www.gleif.org/en/vlei/introducing-the-verifiable-lei-vlei

[89] U.S. Consumer Product Safety Commission, Update: Certificates of Compliance and eFiling, accessed 9 November 2026. https://www.cpsc.gov/Business--Manufacturing/Business-Education/Business-Guidance/Certificates

[90] AERO-CORE research project, public website https://aero-core.com/ (not yet active on 8 November 2026).

[91] ASTM F3269-21, Standard Practice for Methods to Safely Bound Behavior of Aircraft Systems Containing Complex Functions Using Run-Time Assurance, 2021.

[92] EASA, Artificial Intelligence Concept Paper Issue 2: Guidance for Level 1 & 2 Machine Learning Applications, March 2024. https://www.easa.europa.eu/en/document-library/general-publications/easa-artificial-intelligence-concept-paper-issue-2

[93] EASA, Concept Paper: Guidance for Level 1 & 2 Machine Learning Applications, proposed Issue 02, February 2023, anticipated means of compliance SRM-02-1, p. 100. https://www.easa.europa.eu/sites/default/files/dfu/easa_concept_paper_guidance_for_level_1and2_machine_learning_applications_proposed_issue_02_feb2023.pdf

[94] NASA/CR-2020-220586, https://ntrs.nasa.gov/citations/20200003114 ; NASA/CR-20210026909, https://ntrs.nasa.gov/citations/20210026909

[95] A. Kani, NVIDIA Launches NVIDIA Halos, a Full-Stack, Comprehensive Safety System for Autonomous Vehicles, NVIDIA Blog, 18 March 2025. https://blogs.nvidia.com/blog/halos-safety-system-autonomous-vehicles/. See also NVIDIA, Robotaxi and Autonomous Vehicle Safety, "Safety By Layer", accessed November 2026, https://www.nvidia.com/en-us/ai-trust-center/halos/autonomous-vehicles/, and S. Hariharapura Sheshadri et al., Inside NVIDIA Halos for Robotics, NVIDIA Technical Blog, 22 June 2026, https://developer.nvidia.com/blog/inside-nvidia-halos-for-robotics-a-full-stack-functional-safety-system-for-physical-ai/.

[96] D. Schneider, M. Trapp, Conditional safety certification of open adaptive systems, ACM TAAS, 2013. https://doi.org/10.1145/2491465.2491467

[97] M. Trapp, D. Schneider, G. Weiß, Towards safety-awareness and dynamic safety management, EDCC 2018. https://doi.org/10.1109/EDCC.2018.00027

[98] E. Troubitsyna, I. Alvarez, P. Koopman, M. Trapp, Methods and Tools for the Engineering and Assurance of Safe Autonomous Systems, Dagstuhl Reports 14(4), 2024. https://doi.org/10.4230/DagRep.14.4.23

[99] R. Calinescu, D. Weyns, S. Gerasimou, M. U. Iftikhar, I. Habli, T. Kelly, Engineering Trustworthy Self-Adaptive Software with Dynamic Assurance Cases, IEEE Transactions on Software Engineering 44(11), 2018, 1039–1069.

[100] D. Dalrymple et al., Towards Guaranteed Safe AI, arXiv:2405.06624, 2024.

[101] Schubert, Ibrahim, Breitmoser-Widdecke, Durak, An Ontology-based Automation Framework for Continuous Compliance of Airborne Software, DASC 2025, DOI 10.1109/DASC66011.2025.11257376.

[102] F. de Grancey and A. Audouy, Towards the Deployment of Knowledge Based Systems in Safety-Critical Systems, ESWC 2023 Satellite Events, 186–190. https://doi.org/10.1007/978-3-031-43458-7_35

[103] RTCA DO-178C / EUROCAE ED-12C, Software Considerations in Airborne Systems and Equipment Certification, 2011. See FAA Advisory Circular 20-115D for its recognition as an acceptable means of compliance. https://www.faa.gov/airports/resources/advisory_circulars/index.cfm/go/document.information/documentNumber/20-115D

[104] SAE International, Guidelines for Development of Civil Aircraft and Systems, SAE ARP4754B, revised 20 December 2023. https://doi.org/10.4271/ARP4754B

[105] Spherity Research, PACE: Verifiable Safety and Cyber Evidence Graphs for Physical AI, 9 September 2026. https://spherity.github.io/spherity-research/physical-ai-compliance-evidence-pace.html

[106] A. Desai et al., SOTER: A Runtime Assurance Framework for Programming Safe Robotics Systems, arXiv:1808.07921.

[107] U. Mehmood et al., The Black-Box Simplex Architecture for Runtime Assurance of Autonomous CPS, arXiv:2102.12981.

[108] J. J. Carroll, Signing RDF graphs, ISWC 2003, LNCS 2870, 369–384.

[109] G. Tummarello, C. Morbidoni, P. Puliti, F. Piazza, Signing individual fragments of an RDF graph, WWW 2005 (posters), 1020–1021. https://archives.iw3c2.org/www2005/proceedings/docs/p1020.pdf

[110] C. Sayers, A. H. Karp, Computing the digest of an RDF graph, HP Laboratories Technical Report HPL-2003-235(R.1), 2004.

[111] A. Kasten, A. Scherp, P. Schauß, A framework for iterative signing of graph data on the web, ESWC 2014, LNCS 8465, 146–160.

[112] A. Hogan, Canonical forms for isomorphic and equivalent RDF graphs: algorithms for leaning and labelling blank nodes, ACM Transactions on the Web 11(4), 2017, article 22.

[113] T. Kuhn, M. Dumontier, Trusty URIs: Verifiable, Immutable, and Permanent Digital Artifacts for Linked Data, ESWC 2014, LNCS 8465, 395–410. https://arxiv.org/abs/1401.5775

[114] J. J. Carroll, C. Bizer, P. Hayes, P. Stickler, Named graphs, provenance and trust, WWW 2005, 613–622.

[115] C. Bizer, R. Cyganiak, Quality-driven information filtering using the WIQA policy framework, Journal of Web Semantics 7(1), 2009, 1–10. https://doi.org/10.1016/j.websem.2008.02.005

[116] W3C, PROV-O: The PROV Ontology, Recommendation, 30 April 2013. https://www.w3.org/TR/prov-o/

[117] D. Artz, Y. Gil, A survey of trust in computer science and the Semantic Web, Journal of Web Semantics 5(2), 2007, 58–71.

[118] C. Ellison, B. Frantz, B. Lampson, R. Rivest, B. Thomas, T. Ylonen, SPKI Certificate Theory, IETF RFC 2693, September 1999. https://www.rfc-editor.org/rfc/rfc2693

[119] W. E. Vesely, F. F. Goldberg, N. H. Roberts, D. F. Haasl, Fault Tree Handbook, NUREG-0492, US Nuclear Regulatory Commission, 1981.

[120] P. Buneman, S. Khanna, W.-C. Tan, Why and Where: A Characterization of Data Provenance, ICDT 2001, LNCS 1973, 316–330.

[121] T. J. Green, G. Karvounarakis, V. Tannen, Provenance Semirings, PODS 2007, 31–40.

[122] The Semantic Supply Chain: A Comparative Risk Analysis of Credential Meaning Distribution in W3C VCDM, SD-JWT VC, and ISO mdoc, 7 May 2026 (unreviewed working paper, full text unavailable for verification). https://hackmd.io/@KsjE2xL6Q_CAsVkYWt58iA/S17wr3tR-e

[123] L. Pike, A. Goodloe, R. Morisset, S. Niller, Copilot: A Hard Real-Time Runtime Monitor, RV 2010, LNCS 6418, 345–359.

[124] M. Alshiekh, R. Bloem, R. Ehlers, B. Könighofer, S. Niekum, U. Topcu, Safe Reinforcement Learning via Shielding, AAAI 2018. https://doi.org/10.1609/aaai.v32i1.11797

[125] NIST, Module-Lattice-Based Digital Signature Standard (ML-DSA), FIPS 204, 13 August 2024. https://csrc.nist.gov/pubs/fips/204/final

[126] M. Jones, J. Bradley and N. Sakimura, JSON Web Signature (JWS), IETF RFC 7515, May 2015, Section 7.2. https://www.rfc-editor.org/rfc/rfc7515.html

[127] J. Schaad, CBOR Object Signing and Encryption (COSE): Structures and Process, IETF RFC 9052, August 2022, Section 4. https://www.rfc-editor.org/rfc/rfc9052.html

[128] W3C, Verifiable Credentials Working Group Charter, March 2026. https://www.w3.org/2026/03/vc-wg-charter.html

[129] ETSI, ESI Drafts under Public Review, entry for EN 319 482-3 v0.0.7, accessed 7 November 2026. https://portal.etsi.org/TB-SiteMap/ESI/ESI-Drafts-under-Public-Review

[130] Digital Bazaar, rdf-canonize implementation documentation, Introduction. https://github.com/digitalbazaar/rdf-canonize#introduction