

Prove, Don't Disclose

Verifiable CBAM Compliance Across Industrial Supply Chains Using
Rules as Code and Zero-Knowledge Proofs

Yiu Ming Patrick Ma

Keldrion, LLC (Delaware, USA)

ORCID: [0009-0008-9061-6859](https://orcid.org/0009-0008-9061-6859)

Kaimeter Working Paper 2026-01

Version 1.1 · September 2026

DOI: [10.5281/zenodo.22740284](https://doi.org/10.5281/zenodo.22740284)

Text © 2026 Yiu Ming Patrick Ma, licensed under the [Creative Commons Attribution 4.0 International
licence \(CC BY 4.0\)](https://creativecommons.org/licenses/by/4.0/).

Code licensed under the [Apache License 2.0](https://www.apache.org/licenses/LICENSE-2.0/).

Keywords: carbon border adjustment mechanism, CBAM, rules as code, computational law, zero-knowledge proofs, embedded emissions, steel, aluminium, cement, fertilisers, downstream goods, supply-chain privacy, data sovereignty, open source

Abstract

The European Union’s Carbon Border Adjustment Mechanism (CBAM) entered its definitive period on 1 January 2026. From 2027, EU importers of cement, iron and steel, aluminium, fertilisers and hydrogen must surrender certificates for the emissions embedded in their imports; a legislative package still awaiting Parliament’s plenary mandate, and therefore not yet in trilogue, would extend the mechanism from 2028 to several hundred downstream steel- and aluminium-containing products and commit the Commission to reviewing the product list annually. Embedded emissions must either be verified actual values supplied by the non-EU producer — and, for complex goods, by that producer’s own suppliers — or punitive default values carrying a mark-up that rises from 10 % to 30 % over 2026–2028 for most sectors.

The Commission’s own Operators Portal already addresses the single-installation case: an operator registers once and its verified embedded emissions are available to every declarant, as the number and not the activity data. Three gaps remain. The portal centrally holds each installation’s figure in an EU-operated registry; it is per installation, so it cannot assemble a complex or downstream good’s emissions from precursor values held by independent suppliers; and it transports a number without the rule version that produced it.

This paper addresses those three gaps by combining two techniques not previously applied together to border-carbon compliance. First, **rules as code**: the CBAM methodology (Regulation (EU) 2023/956 Annex IV, as implemented by Implementing Regulation (EU) 2025/2547, with default values, benchmarks and mark-ups) is encoded per sector as an executable, tested, versioned *rule bundle* whose cryptographic hash identifies exactly which rules were applied. Second, **zero-knowledge proofs**: a producer executes the codified methodology over its private activity data and attested inputs and produces a succinct proof that the declared figure is the correct output of the identified rule version — without revealing the data. Proofs compose recursively along the supply chain, so a manufacturer of a complex or downstream good can incorporate precursor emissions from installations it never sees.

We describe the rule-bundle and proof formats, the trust anchors connecting proofs to the accredited verification the Regulation requires, deployment models including fully air-gapped operation for producers in jurisdictions with cross-border data-transfer restrictions, and a worked scenario following metal from a smelter through an extruder and a downstream manufacturer to an EU declaration. Aluminium is used as the running example; the architecture is sector-agnostic by construction, and we set out how it applies to steel, cement, fertilisers, hydrogen and the proposed downstream scope. We are explicit that the approach **complements rather than replaces** accredited verification, and we state limitations, open questions and a roadmap. A reference implementation is being developed as open source under Apache 2.0.

1. Introduction: the residual disclosure problem

CBAM is the first border-carbon regime to move from reporting to financial liability at scale. It converts the process emissions of a non-EU installation into a customs-adjacent cost borne by the EU importer, and it does so across sectors — cement, iron and steel, aluminium, fertilisers, hydrogen and electricity — that together account for a large share of the EU’s imported industrial carbon.

The Regulation places the legal obligation on the **authorised CBAM declarant** (the importer or its indirect customs representative). The declarant must declare embedded emissions per

tonne of goods and surrender a corresponding number of certificates. But the declarant does not operate the steel mill, the smelter, the kiln or the ammonia plant. The data required to compute actual embedded emissions belong to the **operator** of a third-country installation — and, for the majority of traded goods, to that operator’s own upstream suppliers.

The Regulation anticipates this. Recital 48 provides that operators of third-country installations should be able to register in the CBAM Registry and make their **verified embedded emissions** available to authorised declarants; Article 10(7) lets an operator disclose its verification information to a declarant, which Article 8(2) permits the declarant to use; and the CBAM Operators Portal implements exactly this — one registration and one verified figure per installation, reachable by every EU declarant through an operator identifier. What travels is the specific embedded emissions, not the operator’s activity data, which Article 10(5)(c) leaves with the installation. For a single installation, “verify once, share many times” is therefore already the law, and this paper does not claim it as a contribution.

The portal leaves three gaps, and it is these — not the single-installation flow — that this paper addresses.

1. **Central custody.** The Registry is established and operated by the Commission, and its information is made available automatically to customs and competent authorities (Art. 14(1)); the Operators Portal is the “unique entry point” to it (IR 2024/3210, Art. 10, as amended by IR 2025/2550). An operator whose jurisdiction restricts cross-border data transfer, or which does not wish to lodge its figure with an EU body, has no way to show a verified number to a buyer except by registering centrally. Commercial sensitivity and legal restriction bite here, at central registration, rather than at delivery to the buyer: an EU importer is frequently a competitor, and producers in China — the largest source of the goods in scope — operate under the Personal Information Protection Law, the Data Security Law and cross-border transfer rules that make releasing operating data to a foreign or central counterparty a compliance question in its own right.
2. **Composition across tiers.** The portal is per installation. It shares one installation’s figure; it does not compute a complex or downstream good’s embedded emissions from precursor values held by a chain of independent operators, nor carry a value up a chain. A downstream manufacturer still needs each precursor’s SEE, each upstream operator would have to register separately, and for lower tiers the manufacturer may not know whom to ask.
3. **Rule identity.** The portal transports a number, not the rule version that produced it. As the methodology, default values and scope change — several times in 2026 alone — a number detached from its rule version cannot be re-verified or compared.

The fallback — default values — is deliberately punitive. Under Implementing Regulation (EU) 2025/2621 (as corrected by IR 2026/1740), default values for cement, iron and steel, aluminium and hydrogen carry a mark-up of 10 % in 2026, 20 % in 2027 and 30 % from 2028; default values for fertilisers carry a mark-up of 1 % in every year. A producer whose real emissions are below the country default is penalised on every tonne for which it fails to demonstrate that fact.

The proposed downstream extension makes composition, not delivery, the binding constraint. A manufacturer of fasteners, wire, structural components or household articles emits little itself; nearly all of its embedded emissions are the emissions of the steel or aluminium it bought. Under the Commission’s approach, its obligation reduces essentially to *mass of CBAM material in the product* $\times$ *precursor embedded emissions*. The portal can deliver a single supplier’s number; it cannot assemble the manufacturer’s own number from a chain of suppliers, some of which it cannot compel and lower tiers of which it cannot even identify.

The residual failure is therefore not that a number cannot reach the importer — the portal can deliver one. It is that delivery depends on central registration in an EU-operated registry, it stops at the installation boundary, and it carries no record of the rules that produced it.

This paper proposes to let the *fact* travel without the *data*, without a central custodian, and with the *rules* attached.

1.1 Contributions

1. **Rules as code for CBAM, per sector.** We specify a versioned, hash-identified *rule bundle* that encodes the CBAM embedded-emissions methodology as executable code with legal citations attached to every parameter and branch, structured so that sector-specific methodology sits on shared cross-sector logic, and we show how rule changes — of which there have already been several in 2026 — and scope changes propagate deterministically (§4).
2. **Zero-knowledge compliance proofs bound to a rule version.** We define a proof statement in which the public inputs include the rule-bundle hash, so a verifier learns not only that “the number is right” but “right *according to which rules*” (§5).
3. **Recursive composition along the supply chain.** We show how a producer of complex or downstream goods incorporates the proven embedded emissions of precursors purchased from other installations without those installations disclosing their activity data beyond what the proof reveals — through as many tiers as the physical chain has (§6).
4. **A trust-anchor model that fits the legal framework.** We map proofs onto the roles the Regulation defines — operator, accredited verifier, declarant, Registry — and state what a proof does and does not establish (§7).
5. **Deployment for data sovereignty.** We describe an air-gapped deployment model, signed offline rule updates, reproducible builds, and an operator-to-declarant flow that requires no registration of the installation or its figure in an EU-operated registry, so that an operator can show a verified number without lodging it centrally (§8).
6. **A specification for an open reference implementation**, with a published test vector for the worked example and a first release to follow (§10, Appendix B).

1.2 Scope

The architecture addresses all goods listed in Annex I of the CBAM Regulation and any goods added to it. Sectors differ in methodology, and the differences are handled in the rule bundle rather than in the architecture:

Sector	Indirect (electricity) emissions included?	Distinctive methodology features	Typical precursor depth
Iron and steel	No (Annex II)	Multiple routes (BF-BOF, DRI-EAF, scrap-EAF); pig iron, DRI and crude steel as precursors; pre-consumer scrap proposed for inclusion	Deep
Aluminium	No (Annex II)	Perfluorocarbons from anode effects; primary vs. secondary	Medium

Sector	Indirect (electricity) emissions included?	Distinctive methodology features	Typical precursor depth
Hydrogen	No (Annex II)	Route-specific (steam methane reforming, electrolysis)	Shallow
Electricity (out of scope here)	No (Annex II, added by Reg. 2025/2083)	Grid regime; embedded emissions by Annex IV point 4.2 default values, or point 5 actual values	None
Cement	Yes	Clinker as precursor; calcination CO ₂ ; electricity emission factors	Shallow, indirect-heavy
Fertilisers	Yes	N ₂ O from nitric acid; ammonia and nitric acid as precursors; distinct 1 % default mark-up	Medium
Downstream goods (proposed, 2028)	Follows precursor	Emissions dominated by precursor content; anti-circumvention origin rules	Whatever the input chain is

Electricity as a CBAM good is out of scope for this paper and the reference implementation: it is a grid-operator regime with its own default-factor logic, not a supply-chain problem. Regulation (EU) 2025/2083 added electricity (CN 2716 00 00) to Annex II, so only its direct emissions are to be taken into account — recital (34) reasons that indirect emissions are not relevant to electricity generation — with embedded emissions determined from Annex IV point 4.2 default values unless the point 5 conditions for actual values are met.

Aluminium is the running example throughout because it is the most demanding single-installation case (PFC emissions require a distinct measurement method) and because it has a deep downstream chain that reaches into the proposed 2028 scope. Nothing in the architecture is specific to it.

1.3 Non-goals

We do not propose to change the methodology, replace accredited verification, or alter the allocation of legal liability. We do not address the UK CBAM or prospective US measures in detail, though §15 notes how the architecture generalises to them.

2. Background: the CBAM definitive regime

This section summarises the legal position as of September 2026. It is not legal advice; the instruments cited (listed with identifiers in Appendix C) are authoritative.

2.1 Timeline that matters for 2026 imports

- **1 January 2026** — definitive period begins; 2026 is the first reporting period for which certificates are owed.
- **1 February 2027** — Member States begin selling certificates on the common central platform; 2026-import certificates priced on 2026 quarterly average EU ETS prices.
- **From 2027, quarterly** — declarants must hold certificates covering at least 50 % of embedded emissions in goods imported year-to-date.
- **30 September 2027** — first annual CBAM declaration (for calendar 2026) and surrender of certificates.
- **1 November 2027** — unused 2026-import certificates cancelled without compensation.

2.2 The downstream extension

On 17 December 2025 the Commission proposed (COM(2025) 989) to extend CBAM from 1 January 2028 to roughly 180 downstream steel- and aluminium-intensive products, to bring pre-consumer scrap into scope, to strengthen anti-circumvention powers (including application of true-origin default values where “slight modification” patterns are found), and to require an annual Commission review adding further downstream goods. The Council adopted a general approach on 12 June 2026 covering roughly 200 products. Parliament’s Environment Committee adopted its position on 6 July 2026 (56–11–12) covering roughly 457 CN codes, having lowered the emissions-intensity threshold used to select goods. Parliament’s plenary mandate was submitted to the 14–17 September 2026 Strasbourg session (following the ENVI Committee’s 56–11–12 adoption on 6 July 2026 covering ~457 CN codes), with trilogue targeted to conclude by Q4 2026.

Whatever the final list, three features are settled across all three institutions and matter for this paper: downstream goods are coming; their embedded emissions will be computed from precursor content; and the scope will thereafter change annually.

2.3 What must be computed

For a good produced in a given installation and reporting period, the **specific embedded emissions** (SEE, in t CO₂e per tonne of good) are the attributed emissions of the production process — direct only for Annex II sectors, direct and indirect otherwise — divided by the activity level, plus, for *complex goods*, the embedded emissions of consumed *precursors*. IR 2025/2547 specifies, among other things:

- the reporting period (Art. 7);
- system boundaries per production route for each sector (Annex II);
- monitoring methods for direct emissions (calculation-based and measurement-based, following the EU ETS Monitoring and Reporting Regulation), and sector-specific methods such as the slope and overvoltage methods for PFCs;
- indirect emissions for cement and fertilisers, using grid or contract-specific electricity emission factors;
- precursor attribution, including weighted averaging when precursors under one CN code arrive from multiple installations or periods (Arts. 13–14);
- treatment of precursors of EU or excluded-territory origin (zero embedded emissions added).

The number of certificates to be surrendered follows from total embedded emissions, less the deduction for a carbon price effectively paid in the country of origin (Article 9, whose implementing rules remained pending at the time of writing), and less the adjustment for free allocation under the EU ETS (Article 31, using the benchmarks in IR 2025/2620).

2.4 Why the rules are a moving target

In the nine months since the definitive period began: default values were published, then corrected retroactively with a structural change to how mark-ups are computed (now within the Registry rather than tabulated), while preserving the sector-dependent mark-up schedule — 10/20/30 % for most sectors, 1 % for fertilisers; methodology guidance was reissued (Guidance Document 3, August 2026); verifier registration rules came into application; the Article 9 implementing act remained in draft; and the scope of the mechanism itself entered legislative revision. The Commission must revise default values and mark-ups by December 2027 and, under the proposal, review the product list every year thereafter. Any compliance system that hard-codes the rules — or the scope — will be wrong within a year. This motivates §4.

3. Problem statement and requirements

We seek a mechanism by which an EU declarant D can obtain, for goods from installation I , a value y (SEE, t CO₂e/t) such that:

- **R1 — Correctness.** y equals the output of the CBAM methodology for I 's sector and production route applied to I 's true activity data for the relevant period.
- **R2 — Confidentiality.** D learns nothing about I 's activity data beyond y and what y implies. Installation identity and the verification report are outside R2: for complex goods they follow the statutory route to the declarant and the competent authority (§7.4).
- **R3 — Rule identity.** D (and any auditor or competent authority) can determine exactly which version of the methodology, default values, benchmarks and parameters produced y .
- **R4 — Anchoring.** y is traceable to inputs the legal framework recognises — in particular, to the report of an accredited verifier where actual values are claimed, under a key resolvable to an accreditation identifier and not chosen by the prover.
- **R5 — Composability.** If I produces complex or downstream goods from precursors supplied by installations $I_1 \dots I_k$, I can incorporate their embedded emissions without $I_1 \dots I_k$ disclosing their data to I , and this holds recursively through any number of tiers.
- **R6 — Sovereignty.** I can run the entire computation on infrastructure it controls, with no network egress, can independently verify that the software it runs is the software whose source it has inspected, and need not register its installation or its verified figure in a centrally operated registry.
- **R7 — Auditability.** Anyone can re-verify y 's proof cheaply, at any later date, against the rule version identified in R3.
- **R8 — Rule and scope agility.** When the rules or the product scope change, the change is published as a discrete, signed, reviewable artifact; existing proofs remain valid *with respect to the rules they cite*; and adding a sector or a CN code does not change the architecture.

No existing approach satisfies all eight. §3.1 explains why this is not a trusted-hardware, multi-party-computation, homomorphic-encryption or clean-room architecture; §3.2 summarises the comparison against R1–R8.

3.1 Adjacent techniques, and why this is not a TEE, MPC, HE or clean-room architecture

A proof is not the only way to compute on data without disclosing it, and a reviewer is entitled to ask why the alternatives were not chosen. Each technique below is real, and none is dismissed outright. Each, however, fails at least one requirement structurally for this setting, where operators are mutually distrusting, sit in several jurisdictions, must be able to operate air-gapped, and must identify the exact legal rule version after the fact.

Trusted execution environments (TEEs) and confidential computing. Technologies such as Intel SGX/TDX, AMD SEV-SNP and Arm CCA execute code in a hardware-isolated enclave and produce a remote attestation binding the running binary. That is a credible way to keep activity data inside an installation, and a TEE can run offline. Its trust root, though, is the silicon vendor and its attestation service rather than mathematics: the operator and the declarant must both accept that root; side-channel and micro-architectural attacks remain a live concern; and in the target jurisdictions a US-origin hardware root is itself a sovereignty problem (§8.1). The object a third party checks is a vendor attestation, not a portable, publicly verifiable proof (R7); and remote attestation binds a *binary*, not the legal citation of a rule version, so R3 and R8 are only as strong as the vendor’s measurement and provisioning story.

Secure multi-party computation (MPC). MPC evaluates a function across parties’ secret inputs with cryptographic rather than hardware trust, and can satisfy R2 strongly. It requires the relevant parties to be online and to interact, which is incompatible with air-gapped operation (R6) and awkward when tiers are unknown to one another and separated by jurisdictions. It yields no compact artifact a regulator can re-verify later (R7) without re-running the protocol, and it carries no notion of a rule version (R3).

Homomorphic encryption (HE/FHE). HE computes on ciphertext, but the result must be decrypted by a key holder, so verification is not public (R7): the verifier must be trusted with the key or the plaintext. FHE also remains too costly for per-shipment use, and it has no built-in rule identity (R3, R8).

Data clean rooms. A clean room is a controlled environment into which each party contributes data under a usage agreement. It centralises custody in the room’s operator, requires data to leave the installation (R6), and substitutes the operator’s access control for a verifiable object (R2, R7). It is a governance mechanism, not a proof.

Environmental Product Declarations and ISO 14064-3 verification. Discussed in §12: these are the incumbent mechanism for a verified figure, and they satisfy R4. They do not bind the figure to an identifiable, machine-checkable rule version (R3), do not adapt deterministically as rules or scope change (R8), and do not compose down a chain as cryptographically verifiable objects (R5).

Trusted intermediary / the CBAM Operators Portal. The portal already satisfies R1, R2 and R4 for a single installation: an operator registers once and its verified embedded emissions are available to declarants, and the declarant sees the figure, not the operator’s activity data (Recital 48; Art. 10(7); Art. 8(2)). It fails R6, because registration and custody are central and the Registry is EU-operated and made available to authorities (Art. 14(1)); R5, because it is per installation and does not compose a chain; and R3 and R8, because it carries a number without the rule version. This is the baseline the architecture is designed against.

Two qualifications, stated plainly. First, the rule-bundle contribution is *separable* from the proof system: a TEE or MPC implementation could in principle bind a rule-bundle hash. The argument for zero-knowledge is not that the others cannot carry rule identity, but that zero-knowledge is the only option here that is simultaneously non-interactive, air-gappable, publicly re-verifiable by any third party, and free of a single hardware or operator trust root — which is what a multi-jurisdiction, mutually distrusting supply chain requires. Second, these techniques are complementary, not excluded: a TEE is a reasonable place to hold an operator’s signing key, and MPC is a reasonable way to aggregate data across an operator’s own sites. The architecture does not forbid them; it declines to make them the trust root.

3.2 Comparison against R1–R8

Legend: ✓ satisfied; ●satisfied only under assumptions (trust in a party or in hardware, connectivity, or re-running a protocol); ×not satisfied.

Approach	R1	R2	R3	R4	R5	R6	R7	R8
Rule bundle + zero-knowledge proofs (this paper)	✓	✓	✓	✓	✓	✓	✓	✓
TEE / confidential computing	✓	●	●	●	●	●	●	●
Secure multi-party computation	✓	✓	×	●	●	×	●	×
Homomorphic encryption	✓	✓	×	●	●	●	×	×
Data clean room	✓	●	×	●	×	×	●	×
EPD / ISO 14064-3 verified declaration	●	●	×	✓	×	●	●	×

Approach	R1	R2	R3	R4	R5	R6	R7	R8
CBAM Opera- tors Portal (incum- bent)	✓	✓	×	✓	×	×	●	×
PCF ex- change format (PACT, Catena- X)	●	●	×	●	●	●	×	×

No row but the first claims all eight. R1 here means the computation is performed correctly on the inputs it is given; the separate question of input *truth* is carried by R4, and every approach ultimately depends on attestation or verification for it (§5.3, §13.1). The matrix is an argument about structure, not a benchmark: where a technique could be strengthened with substantial further engineering, the cell is marked ● rather than ×.

4. Rules as code: the methodology as a versioned executable artifact

4.1 Principle

Rules as code (RaC) is the practice of publishing legislation or regulation as machine-executable logic alongside — not instead of — the natural-language text, so that the logic can be tested, versioned and reused. The OECD’s *Cracking the Code* (2020) and systems such as OpenFisca and Catala established the approach for tax and benefit law. We apply it to a technical emissions methodology, where it has three specific advantages: the methodology is already largely arithmetic; its inputs are already required to be monitored under a documented plan; and the consequences of rule drift are financial and immediate.

4.2 The rule bundle

A **rule bundle** *B* is a self-contained library of source code and parameter data. Cross-sector logic lives once, in `common/`; each sector contributes only what is specific to it. The reference implementation realises the bundle as a Rust crate, `kaimeter-rules`, with no I/O, no floating-point arithmetic and no dependency on the surrounding application — properties chosen so that the same code can be evaluated natively and, under the fixed interpreter of §5.4, proven directly with the bundle supplied as witness.

```
crates/kaimeter-rules/
```

```
  Cargo.toml
```

```
  bundle.json
```

```
      # metadata: jurisdiction, sectors, appliesFrom, legalBasis, sup
```

```

attestation-policy.json      # admitted attester IDs, keys and accreditation IDs (signed; §§
CITATIONS.md                # provision-by-provision mapping to legal text
src/
  lib.rs
  fixed.rs                  # fixed-point arithmetic (§4.4)
  bundle.rs                 # canonical serialisation and hash (§4.3)
  common/
    period.rs               # reporting-period determination (IR 2025/2547 Art. 7)
    precursors.rs           # attribution, weighted averaging, EU-origin zeroing (Arts. 13-
14)
    defaults.rs             # default-value selection (IR 2025/2621 as corrected)
    markups.rs              # sector-dependent mark-up schedule (§4.6)
    indirect.rs             # electricity emission factors, for sectors where indirect coun
    certificates.rs         # Art. 9 deduction, Art. 31 free-allocation adjustment
    scope.rs                # Annex I CN codes -> sector/route, with appliesFrom dates and
  sectors/
    aluminium/
      boundaries.rs          # routes: primary, secondary, downstream processing
      pfc.rs                 # slope and overvoltage methods
    steel/
      boundaries.rs          # routes: BF-BOF, DRI, EAF, rolling/finishing
      scrap.rs               # pre-consumer scrap treatment (proposed)
    cement/
      boundaries.rs          # clinker, cement, calcined clay
      calcination.rs
    fertilisers/
      boundaries.rs          # ammonia, nitric acid, urea, mixed fertilisers
      n2o.rs
    hydrogen/
      boundaries.rs
    downstream/
      content.rs             # precursor-content model for downstream goods (proposed, 2028)
      origin.rs              # anti-circumvention / true-origin rules (proposed)
  parameters/               # data tables, each row with source URL and retrieval date
    gwp.json
    default-values-annex-I.json
    default-values-annex-IV.json
    benchmarks-2025-2620.json
    electricity-factors.json
  tests/
    vectors/<sector>/*.json # test vectors, including Commission guidance examples

```

Every public function carries a structured annotation linking it to the provision it implements:

```

/// CF4 emissions from anode effects, slope method.
///
/// @legal IR (EU) 2025/2547, Annex II, point B.7.1
/// @source http://data.europa.eu/eli/reg_impl/2025/2547/oj
/// @since bundle 2026.1.0
pub fn pfc_slope(input: &SlopeInput, p: &PfcParameters) -> Result<PfcResult, RuleError> { /*

```

Adding a sector means adding a module under `sectors/` and entries in `scope.rs`; adding a CN

code to the downstream list means a row in `scope.rs` with an `appliesFrom` date and a legal citation. Neither touches `common/`, the proof system, or the envelope format of §9. This is the architectural form of requirement R8.

4.3 Version identity

The bundle is identified by a hash over its canonical serialisation:

$$h_B = H(\text{canon}(B))$$

where H is a collision-resistant hash (SHA-256 for the bundle; a proof-system-native hash such as Poseidon for the in-circuit commitment) and `canon` is a deterministic serialisation — sorted paths, normalised line endings, non-semantic files excluded. h_B appears as a **public input in every proof** (§5). Two parties disagreeing about a result can immediately determine whether they are disagreeing about the *inputs* or about the *rules*.

Bundles are versioned semantically — `2026.1.0`, `2026.1.1` — and each bundle’s metadata records what it supersedes and why, with the legal instrument that motivated the change. The July 2026 correction to default values is bundle `2026.2.0`, applying from 1 January 2026, superseding `2026.1.x`. When the downstream extension is adopted, its CN codes enter `scope.rs` with `appliesFrom: 2028-01-01` in a `2027.x` bundle — visible, reviewable and testable a year before they bite. A proof generated under an earlier bundle remains a valid proof of what the rules *were understood to be*; the operator can re-prove under the new bundle with the same private inputs, and the two proofs together document the change.

4.4 Determinism and arithmetic

Zero-knowledge proof systems operate over finite fields; the methodology is specified in decimal. We therefore fix:

- a **fixed-point scale** $s = 10^6$ for all quantities, i.e. every value x is represented as the integer $\lfloor x \cdot s \rfloor$, held in a 128-bit signed integer;
- **rounding rules** at each step that mirror the Regulation’s stated precision, with round-half-to-even where the Regulation is silent;
- an **evaluation order** defined by the bundle, not by the host language.

The rules crate forbids floating-point arithmetic entirely. The reference implementation will ship a plain evaluator (native Rust) and, in a subsequent release, a provable evaluator — initially the fixed interpreter of §5.4, with the bundle supplied as witness rather than compiled in — together with a **differential test suite** asserting that the two agree on every test vector and on randomly generated inputs. Independent third parties are invited to maintain their own evaluators against the same vectors.

4.5 Worked example: primary aluminium with PFC emissions

Consider a smelter using prebake cells, monitored under the slope method. Inputs for the reporting period:

Symbol	Meaning	Illustrative value
Pr_{Al}	Primary aluminium produced	100 000 t
E_{CO_2}	Direct CO ₂ from anode consumption, fuels, flue-gas treatment (mass balance)	155 000 t
AEM	Anode-effect minutes per cell-day	0.25
SEF_{CF_4}	Slope emission factor, (kg CF ₄ / t Al) per (AE-min / cell-day)	0.143
$F_{C_2F_6}$	Weight fraction C ₂ F ₆ / CF ₄	0.121
$GWP_{CF_4}, GWP_{C_2F_6}$	Global-warming potentials carried in the bundle (AR5 values as adopted in the EU ETS MRR: 6 630 and 11 100)	—

$$E_{CF_4} = AEM \times \frac{SEF_{CF_4}}{1000} \times Pr_{Al} = 0.25 \times \frac{0.143}{1000} \times 100\,000 = 3.575 \text{ t CF}_4$$

$$E_{C_2F_6} = E_{CF_4} \times F_{C_2F_6} = 3.575 \times 0.121 = 0.433 \text{ t C}_2\text{F}_6$$

$$E_{PFC} = E_{CF_4} \cdot GWP_{CF_4} + E_{C_2F_6} \cdot GWP_{C_2F_6} \approx 23\,702 + 4\,802 = 28\,504 \text{ t CO}_2\text{e}$$

$$SEE_{Al} = \frac{E_{CO_2} + E_{PFC}}{Pr_{Al}} = \frac{155\,000 + 28\,504}{100\,000} = 1.835 \text{ t CO}_2\text{e / t}$$

(Figures are illustrative, within typical ranges for modern prebake smelters; they are not measurements from any installation. Intermediate values are rounded for display; at the bundle's fixed-point precision the result is 1.835038, as given in the test vector in Appendix B. Authoritative GWP values are those in the applicable Annex, carried in *gwp.json*.)

Every line above will correspond to a function in `sectors/aluminium/`, and this example is the first published test vector for the bundle. The same pattern — sector module computes attributed emissions; `common/` handles period, precursors, defaults and certificates — applies to a BF-BOF steel mill (`sectors/steel/boundaries.rs`, with pig iron and crude steel as intermediate precursors), a clinker kiln (`calcination.rs` plus `indirect.rs`), or a nitric-acid plant (`n2o.rs`).

4.6 Default values and mark-ups as rules

Default-value selection has several branches — country listed or not, value present or “—”, precursor of unknown origin (Annex IV) — and the mark-up depends on both the sector and the year of import:

$$DV_{\text{applied}}(c, g, Y) = DV_{\text{total}}(c, g) \times (1 + m_{s(g)}(Y))$$

$$m_s(Y) = \begin{cases} 0.10 & Y = 2026 \\ 0.20 & Y = 2027 \\ 0.30 & Y \geq 2028 \end{cases} \quad \text{for } s \in \{\text{cement, iron and steel, aluminium, hydrogen}\}; \quad m_{\text{fertilisers}}(Y) = 0.01$$

The fertiliser exception is exactly the branch a hand-maintained spreadsheet omits. The proposed anti-circumvention rules add a further branch: where goods are found to have undergone only slight modification in an intermediate country, the default of the *true origin* applies. Encoding this as a rule with the history attached is exactly the case where a spreadsheet fails silently and a versioned bundle does not.

5. Zero-knowledge compliance proofs

5.1 What is proven

A zero-knowledge proof allows a prover to convince a verifier that a statement is true without revealing why. Here the statement is:

“I know private inputs w such that running the rule bundle identified by h_B , for the context π_{ctx} , on w yields the public output y , and every input in w that the bundle requires to be attested carries a valid attestation from an attester whose identity and public key are admitted by the attestation policy h_P and resolvable to an accreditation identifier, over a tuple that binds the installation commitment, the reporting period, the CN code and route, the rule-bundle hash h_B , the attester identity, and a root over the attested inputs.”

Formally, with public inputs $x = (h_B, h_P, y, \pi_{\text{ctx}})$ and private witness w :

$$\mathcal{R}(x, w) \iff h_B = H(\text{canon}(B)) \wedge F(B, w) = y \wedge \text{Attested}_{h_P}(w, \pi_{\text{ctx}}, h_B) \wedge \text{Context}(w, \pi_{\text{ctx}})$$

where B is the rule bundle carried in the witness, F is the fixed bundle interpreter of §5.4 (so h_B is recomputed in-circuit rather than trusted from the prover), and Attested_{h_P} requires, for each input the bundle marks as attested, that

- the attester identifier appears in the attestation policy identified by h_P , and its signing key resolves from that policy to an accreditation identifier under the verifier-registration framework (Delegated Regulation (EU) 2025/2551; IR 2025/2550);
- the signature verifies over the canonical tuple (installation commitment, period, CN code, route, h_B , attester identity, and a root r_w over the attested inputs; and
- that tuple is bound into π_{ctx} , so an attestation cannot be replayed for a different installation, period, good, route or rule version.

π_{ctx} also carries a commitment to the installation identifier. The verifier learns y and the context; it does not learn any component of w .

The difference from a prover-supplied commitment is deliberate. There is no c_A chosen by the prover: the attester keys and accreditation identifiers are public and are *resolved by the verifier from h_P* . A producer cannot generate a keypair, commit it, self-sign its inputs and satisfy the predicate, because that key appears in no policy the verifier accepts, and a policy the producer invents has a different hash. The policy is distributed as a signed `attestation-policy.json` alongside the rule bundles (§9.1), and h_P is a public input, so a proof made under one policy is not accepted under another.

For R4 to hold, the policy must be authored by a party independent of the prover — in practice, derived from the verifier registrations under the accreditation framework rather than maintained by the software vendor. The proof system cannot enforce that independence; it can only make the choice of policy explicit and checkable.

5.2 Public and private inputs

Public (in the proof)	Private to the proof (not revealed by it)
Rule-bundle hash h_B	Production tonnage, activity data, fuel and material flows
Output y (SEE) and, optionally, coarse aggregates the operator <i>chooses</i> to reveal	Sector-specific measurements (anode-effect minutes, N ₂ O concentrations, clinker factors, electricity consumption)
Context: CN code, sector, route, period, installation commitment	Precursor purchase quantities and supplier identities
Attestation-policy hash h_P (which resolves the admitted attester IDs and keys) and the attested-input root r_w	The attestations themselves — signatures, meter data, laboratory results and the verifier’s working papers — as distinct from the statutory contents of the verification report
Hashes of incorporated upstream proofs (§6)	Upstream proofs’ private inputs (never known to this prover either)

“Private to the proof” is not a claim that a value never leaves the operator. The verification report, and the installation identifiers it must contain for complex goods, follow their own statutory route to the declarant and the competent authority, and raw activity data remains with the operator (which Art. 10(5)(c) requires it to retain); §7.4 sets out the boundary precisely.

5.3 Trust anchors

A proof establishes that the *computation* is correct. It cannot establish that the *inputs* are true — an installation that lies to its own software obtains a valid proof of a false number. The architecture therefore requires that inputs which determine the result be **attested**, and the proof checks the attestations. The framework recognises three classes:

1. **Accredited-verifier attestation.** Where actual values are claimed, Article 8 requires verification by an accredited verifier. The verifier, after its site visit and review, signs a structured attestation of the verified inputs (or of the verified result and the bundle hash used), under a key that resolves through the attestation policy to its accreditation identifier. This is the anchor the legal framework itself recognises, and it is the primary anchor in this architecture. Because the key is admitted by the policy and the attestation binds the full tuple (§5.1), a self-signed attestation cannot stand in for it.

2. **Instrument attestation.** Signed meter or laboratory data. Supplementary evidence, and useful in the interim before verifier capacity exists; not a substitute for anchor 1.
3. **Upstream proof.** For precursors, the attestation *is* another proof (§6).

We call the resulting pattern **verify once, prove many times**. It should be stated plainly that the pattern is not itself new: the Operators Portal already gives one installation’s verified figure to multiple declarants (Recital 48; Art. 10(7); Art. 8(2)). What the proof adds is that the sharing need not pass through the EU-operated Registry, and that the same attestation can be composed into a downstream good’s proof — so a verified figure can travel both *laterally*, to any declarant, and *upward*, into a complex good, without the operator registering it centrally.

5.4 Proof systems and engineering choices

The design is proof-system-agnostic. The intended primary target is a **general-purpose zero-knowledge virtual machine** (zkVM), which proves the correct execution of an ordinary compiled program. We adopt a specific pattern — **a fixed interpreter with the bundle supplied as witness**, not a rule set compiled into the guest — because the naive alternative does not survive the deployment requirements.

If the rule bundle were compiled *into* the guest, every rule change would change the guest image and therefore the verification key a declarant must pin. An operator that rebuilds the guest locally — which air-gapped, reproducible deployment invites, and which R6 requires it to be able to do — would produce a different image whenever the rules or the toolchain changed. The declarant would then have to track and accept an unbounded set of images, and “the bundle hash and the guest-program hash identify the same artifact” would hold only for a single coordinated build, not for a multi-jurisdiction deployment in which each party builds for itself.

Instead, the guest is a **fixed, versioned rule-bundle interpreter** with a stable image identifier. It takes the bundle B as a witness, recomputes $h_B = H(\text{canon}(B))$ **inside the proof**, checks the recomputed value against the public input h_B , and only then evaluates F over the private inputs w using the rules in B . The bundle travels as data; the interpreter’s image ID changes only when the interpreter changes, not when a rule, a default value or a CN code changes. This gives the properties claimed:

- **Rule identity (R3).** The public h_B is not merely asserted by the prover; it is recomputed in-circuit from the bundle witness, so the rules actually evaluated are the rules whose hash appears in the envelope.
- **Regulatory agility (R8).** A new sector or CN code is new *data* for the same interpreter; no new circuit and no new guest image is required.
- **Sovereignty and reproducibility (R6).** The operator can rebuild the interpreter from published source and obtain a byte-identical image ID, and the declarant pins one small set of interpreter IDs rather than an image per rule version.

The cost is that canonical hashing and rule dispatch are proven rather than native, and the bundle witness increases proving time and proof size. The canonical hash must be computable inside the proof system: SHA-256 over the canonical serialisation is available but expensive in most zkVMs, so the bundle may instead be committed with a proof-system-friendly hash over the same serialisation, with the two constructions documented and conformance-tested. A hand-written circuit remains a fallback where proving cost dominates; the same interpreter contract — bundle as witness, h_B recomputed in-circuit, stable key — applies to it.

Two points should be stated plainly. First, **no prover exists at the time of publication**; the first interpreter guest build is planned for the second release of the reference implementation

(§10). Second, based on published benchmarks for comparable workloads, proving time for a single installation’s methodology at the fixed-point precision above, together with in-circuit bundle hashing, is expected to be seconds to low minutes on commodity hardware, with verification in milliseconds — neither a bottleneck for annual or per-shipment use. These expectations will be replaced by measurements in a later version of this paper.

6. Multi-tier supply chains and recursive composition

6.1 The paradox, one tier up

Most CBAM goods that cross the EU border are **complex goods**: steel sections, aluminium profiles, sheet, wire, castings, cement from imported clinker, mixed fertilisers from imported ammonia — each embodying precursor goods produced elsewhere, often by a different company. IR 2025/2547 requires the producer of a complex good to include the embedded emissions of its precursors, with weighted averaging where precursors under one CN code come from several installations or periods (Arts. 13–14).

This presents the disclosure paradox a second time, one tier upstream: the extruder needs the smelter’s SEE; the re-roller needs the slab producer’s; the nitrate plant needs the ammonia plant’s. None of those suppliers wants to give a customer with negotiating leverage its process data.

6.2 Proofs compose

Let an upstream installation produce a proof π_1 for public output y_1 under bundle h_B . The downstream producer’s proof π_2 takes (π_1, y_1) as *private* inputs, verifies π_1 **inside its own computation**, uses y_1 in the precursor-attribution step, and outputs y_2 . The downstream producer learns nothing about its supplier’s *data* beyond y_1 ; the EU declarant verifying π_2 learns only y_2 and — if the producer chooses to expose it — the hash of π_1 as evidence that a precursor proof was incorporated. This is a statement about the proof, not about the compliance record: for complex goods, the verification report must still identify the installations at which the precursors were produced (Annex VI, point 2(k)(iii)), and that identifier travels by the report rather than by the proof (§7.4).

$$\pi_2 : \exists w_2, \pi_1, y_1 . \text{Verify}(\pi_1, (h_B, y_1, \dots)) \wedge F_{h_B}(w_2 \| y_1) = y_2$$

Weighted averaging across k suppliers generalises directly: π_2 verifies $\pi_{1,1} \dots \pi_{1,k}$ and computes the mass-weighted average per Art. 14(2), or — where the operator can evidence single-supplier use per Art. 14(3) — the specific value. Precursors of EU origin, or from excluded territories, contribute zero embedded emissions; the bundle encodes this branch and the proof can attest the origin claim via a customs document hash or signed supplier declaration.

6.3 Depth

Because π_2 is itself a proof of the same form, a third tier composes over it identically, and so on. The chain ore $\rightarrow$ pig iron $\rightarrow$ crude steel $\rightarrow$ hot-rolled coil $\rightarrow$ cold-rolled sheet $\rightarrow$ stamped component $\rightarrow$ assembled article is five compositions, each performed by the party that holds the relevant physical inputs, each learning only the SEE of the good it bought. No party ever holds the full chain’s data — including the platform provider, who holds none of it.

6.4 Downstream goods and the 2028 extension

The proposed downstream extension is where this matters most. For a downstream good, the Commission’s approach reduces embedded emissions to the CBAM-material content of the product multiplied by the embedded emissions of that material, plus (where material) the producer’s own processing emissions. For a manufacturer of fasteners, hardware, structural parts or household articles this has three consequences:

- **Its number is almost entirely someone else’s number.** A bolt maker’s own emissions per tonne are small; the steel’s are not. Without an actual value for the steel, the bolt maker’s importer defaults — with the 30 % mark-up from 2028.
- **The someone else is often two or more tiers away.** A stamping shop buys sheet from a service centre that buys coil from a mill. The stamping shop may not know which mill. A proof passed down the chain carries the mill’s SEE without carrying the mill’s identity — though for that complex good the verification report must identify the installation that produced the precursor (Annex VI, point 2(k)(iii)); the proof relieves the *customer’s* operational awareness, not the *record’s* completeness (§7.4).
- **Anti-circumvention and proofs point the same way.** The proposal’s anti-circumvention powers target goods whose declared origin conceals their true origin. A proof bound to an installation commitment and a verifier attestation *is* evidence of true origin at the material level. The architecture and the enforcement objective are aligned, not in tension.

The bundle handles downstream goods in `sectors/downstream/content.rs`: the rule takes the good’s CN code, the mass of each CBAM precursor material it contains (an attested input — bill of materials, weighed inputs, or a Commission-specified content coefficient where actual content is not monitored), the corresponding upstream proofs, and the producer’s own attested process emissions, and returns the SEE. The product list itself is data in `scope.rs`, not logic; when trilogue settles the list — and each year the Commission revises it — the change is a signed bundle release.

6.5 The proof as a supply-chain currency

The practical consequence is that the **proof format becomes a currency of trust within the supply chain**, independent of the EU declarant. A mill or smelter can issue proofs to every customer at marginal cost; a component maker can differentiate on a low SEE without disclosing its suppliers’ activity data; a trader can pass proofs through without ever holding data it would be liable for protecting; and a manufacturer of downstream goods who today has never heard of CBAM can, in 2028, satisfy its importer by forwarding what its suppliers already produce.

7. Fit with the legal framework

It is essential to be precise about what this architecture is and is not.

7.1 What it does not replace

- **Accredited verification (Art. 8, Annex VI).** Actual values must be verified by an accredited verifier, with a site visit in the terms the accreditation rules require. A zero-knowledge proof is not verification and does not make a verifier unnecessary. It is a mechanism for *transporting* the result of verification without transporting the data. The verifier’s key is admitted only through the attestation policy (§5.1), which resolves it to an accreditation identifier; a self-generated key cannot substitute for it.
- **The declarant’s liability.** Regulation 2025/2083 allows the declarant to delegate *submission* of the declaration, and IR 2025/2550 provides for delegated access to the Registry; the delegating declarant remains responsible for its obligations. A proof does not shift liability. It gives the declarant an audit trail commensurate with the liability it already bears.
- **The CBAM Registry and Operators Portal.** The Registry remains the channel of record, and the declaration is still filed through it. The portal is also the incumbent mechanism by which an operator shares a verified figure with declarants (Recital 48; Art. 10(7)). The architecture does not bypass the Registry; it offers the operator-to-declarant leg without requiring the operator to register centrally, while the declarant files and retains exactly what the Regulation requires (§7.4).
- **Default values.** Where an operator cannot or will not obtain verification, default values apply, with mark-up. The architecture makes the alternative — proving an actual value — available to operators who could not previously do so without disclosure.

7.2 Where it fits

Role	Today	With this architecture
Operator (non-EU installation, any tier)	Registers in the Operators Portal and makes its verified embedded emissions available to declarants (Recital 48; Art. 10(7)); the figure is held centrally in an EU-operated registry, while activity data stays with the operator (Art. 10(5)(c))	Runs the bundle locally; obtains verifier attestation once; issues proofs directly to any number of declarants and downstream producers, without central registration
Accredited verifier	Verifies data; issues report	Additionally signs a structured attestation, under a policy-resolved accredited key, that the proof can check; can rely on the bundle to check arithmetic, concentrating effort on inputs and monitoring, where human judgement matters
Downstream manufacturer	Requests data from suppliers; often cannot obtain it; importer defaults	Receives supplier proofs; composes its own; never sees supplier data

Role	Today	With this architecture
Declarant	Receives data or uses defaults; liable for correctness	Receives y and proof; verifies in milliseconds; declares y ; retains proof, bundle hash, the accepted attestation-policy hash and attested-input root, and the verification report it is required to file and keep
Competent authority / Commission	Reviews declarations; may request evidence	Can re-verify proofs against the published bundle at any time; can publish or endorse bundles as reference implementations

7.3 Evidentiary status

Nothing in the CBAM framework currently accords a cryptographic proof any specific evidentiary status. We do not claim one. We claim that a declarant holding a verifier attestation, a proof bound to that attestation, and a public rule bundle is in a materially stronger evidentiary position than a declarant holding a spreadsheet emailed by a supplier. Whether competent authorities come to *prefer* this form of evidence is an empirical question the coming declaration cycles will answer; §15 proposes engagement to accelerate it.

7.4 Confidentiality, scoped

The paradox this paper addresses is a paradox about **activity data** — production tonnages, fuel and material flows, measured concentrations, purchase quantities, and the cost and technology information they reveal. It is not a claim that the supply chain can be made anonymous. Three features of the framework prevent that, and this architecture does not purport to defeat them:

- **Installation identity travels with the verification report.** In the case of complex goods, Annex VI, point 2(k)(iii) requires the verification report to identify the installations at which the precursor was produced and the actual emissions from its production. The proof envelope carries only a commitment to the installation identifier (§5.1, §9.2); the identifier itself travels in the report, as it does today.
- **The report is part of the declaration.** Article 6(2)(d) places copies of the verification report in the CBAM declaration. The declarant’s obligation to the competent authority is not discharged by holding a proof.
- **The record is retained.** Articles 7(5)–(6) with Annex V, point 2 require the declarant to keep the Annex V records — installation identification, the verification report, the specific embedded emissions and the method used to calculate them — until the end of the fourth year after the declaration is submitted. A proof cannot be substituted for that record.

What survives — and it is the substantive claim — is that the **underlying activity data does not travel**. The framework’s disclosure duty runs to the verification report and the specific embedded emissions, not to the operator’s raw data; Article 10(5)(c) of the Regulation separately obliges the operator to retain its records and a copy of the verification report. It is data of that kind, not the fact of which installation produced a tonne, that the proof withholds. Accordingly:

- Statements such as “no supplier identities” and “no verifier report contents” are not made at the scope the framework permits. The precise formulation is the one used in §8.4: *no activity data, no production or purchase volumes, no measurements*.
- Where the law or a lawful request requires the record, the architecture specifies an explicit **audit mode** rather than leaving an implicit gap. On a request by a competent authority exercising its verification or inspection powers, by an accredited verifier under the accreditation rules, or by a party to whom the operator has granted a contractual audit right, the operator releases the underlying activity data, the attestation, or both directly to the requester. This release is outside the proof system by design: the proof changes the *default* position — no activity data leaves the installation absent such a request — not the operator’s or the declarant’s legal obligations.

8. Deployment models and data sovereignty

8.1 The sovereignty requirement

For a producer in China — or in any jurisdiction with comparable controls — the question “where does my data go?” is prior to every other question. Cross-border transfer of industrial operating data may require security assessment or standard-contract filings; state-owned enterprises face additional internal controls; and commercially, releasing installation data to a foreign vendor is simply unattractive. Any architecture that requires data to leave the installation to be computed upon will not be adopted at scale in the jurisdictions that produce most of the world’s steel, aluminium and cement.

The architecture is therefore designed so that **no raw activity data ever needs to leave the installation** — only the proof, its public inputs and the rule-bundle hash — and so that the operator can verify this claim rather than trust it. The verification report, and the installation identifiers it must contain where they are required, follow the statutory route set out in §7.4.

8.2 Deployment tiers

Tier	Description	Typical user
Air-gapped	Software installed from verified media on a network-isolated host. No outbound connectivity. Rule bundles imported as signed archives via removable media. Proofs exported as files or QR codes.	Producers in restrictive jurisdictions; SOEs; anyone whose IT policy forbids egress
Self-hosted, connected	Same software on the operator’s own infrastructure with outbound access for bundle updates over signed channels. No inbound access.	Producers and downstream manufacturers with standard enterprise IT

Tier	Description	Typical user
Hosted (cloud)	Operated by a service provider for parties who do not wish to self-manage. Appropriate for EU declarants, who must in any case hold the data they declare, and for smaller operators without IT capacity.	EU importers; SMEs

The software is identical across tiers. The air-gapped tier is not a restricted edition; it is the reference configuration.

8.3 Properties the software must have, and how they are demonstrated

- **No telemetry, no licence check, no update check.** Stated in the documentation, enforced in code, verifiable by network monitoring during evaluation.
- **Signed rule bundles.** Each bundle is distributed with a detached signature from the maintainers’ published keys; the software refuses unsigned or mis-signed bundles. Because every proof carries h_B , an offline installation that lags behind a correction cannot silently produce proofs under stale rules without that fact being visible to the verifier.
- **Reproducible builds.** Release binaries are built deterministically so that any party can build from published source and obtain a byte-identical artifact. This converts “trust the vendor” into “verify the build”, which for a US-origin tool deployed in China is a precondition, not a nicety.
- **Software bill of materials and provenance** (SBOM; SLSA-style provenance attestations) published with every release.
- **Platform support** for the Linux distributions in actual use at target installations, including openEuler and Kylin, with offline package mirrors documented.

8.4 What crosses the border

For a shipment from a non-EU manufacturer to an EU importer, the complete cross-border data flow is:

1. Proof π (a few kilobytes);
2. Public inputs: h_B , h_P , y , context (CN code, sector, route, period), attested-input root r_w ;
3. Optionally, hashes of incorporated upstream proofs.

What does **not** cross the border on this path: raw activity data, production volumes, fuel and material flows, measurements, and precursor purchase quantities. What the architecture does not, and cannot, withhold: for complex goods, the verification report that Article 6(2)(d) places in the declaration identifies the installations at which the precursors were produced (Annex VI, point 2(k)(iii)), and the declarant must keep the report, the specific embedded emissions, the installation identification and the method used for four years (Art. 7(5)–(6) and Annex V, point 2). Those artifacts follow their existing statutory route — operator and verifier to declarant, declarant to the competent authority — and the proof does not carry them. Where a competent authority, an accredited verifier or a contractual auditor makes a lawful request, the operator can release the underlying activity data or attestation through the audit mode of §7.4.

9. Interoperability: proof and rule-bundle specification

For the approach to succeed it must not be proprietary. We propose — and the reference implementation adopts — open specifications for two artifacts.

9.1 Rule-bundle specification (sketch)

- Canonical serialisation for hashing (deterministic ordering, normalised line endings, excluded paths listed in `bundle.json`).
- Required metadata: `jurisdiction`, `sectors[]`, `appliesFrom`, `appliesTo`, `legalBasis[]`, `supersedes`, `changelog`.
- Required annotation schema for legal citations on every exported function and every parameter table row.
- Required test-vector format per sector, so independent evaluators can be conformance-tested.
- Signature format (detached, over the canonical hash).
- A signed `attestation-policy.json`, distributed alongside bundles and identified by its own hash h_P : for each admitted attester, the attester identifier, its role (accredited verifier or instrument), the public key or key fingerprint, the accreditation identifier under Delegated Regulation (EU) 2025/2551, and a validity window. Proofs cite h_P , and a verifier accepts only policies it recognises; the policy's accreditation identifiers are checkable against the verifier registrations published under IR 2025/2550.

9.2 Proof envelope (sketch)

```
{
  "version": "1",
  "proofSystem": "<system/version>",
  "ruleBundleHash": "sha256:...",
  "publicInputs": {
    "output": { "see_tco2e_per_t": "1.835038" },
    "context": { "cnCode": "7601", "sector": "aluminium", "route": "primary", "period": "2025-01-01",
    "attestationPolicyHash": "sha256:...",
    "attestedInputRoot": "sha256:...",
    "attesters": [{ "id": "...", "accreditationId": "..." }],
    "incorporatedProofs": ["..."]
  },
  "proof": "base64...",
  "verifierHints": { "verificationKeyHash": "..." }
}
```

A declarant's software verifies the envelope, checks h_B against the list of bundles it accepts, checks that h_P names an attestation policy it recognises, resolves each attester's key from that policy, and confirms the tuple bound into the attestation — installation, period, CN code, route, h_B , attester ID, attested-input root — matches the envelope's context. It then records the envelope as evidence. The envelope is identical for a steel coil, a bag of cement or a box of fasteners; only the context differs.

9.3 Relationship to existing standards

Product carbon footprint exchange formats (e.g., WBCSD PACT) and industry data-space initiatives (e.g., Catena-X) define how a footprint value travels between systems. The proof envelope is complementary: it can be carried as an attachment to such a record, adding proof of correctness and rule identity to a format that otherwise transports an unverified assertion. We would welcome standardisation of the envelope within one of these bodies rather than as a stand-alone specification.

10. Planned reference implementation

A reference implementation, **Kaimeter**, is being developed as open source under Apache 2.0 at <https://github.com/kaimeter/kaimeter>. At the time of this preprint it is not yet released; the description below is a specification and a commitment, not a report of results. A later version of this paper will replace this section with the status of a tagged release.

Structure. A Rust workspace of three crates:

- **kaimeter-rules** — the rule bundle of §4: `common/` logic, per-sector modules, parameter tables, test vectors, canonical hashing. Pure, deterministic, `no_std`-compatible, no I/O, no floating-point arithmetic, no dependency on the other crates. This crate is the artifact this paper describes.
- **kaimeter-core** — declarant and operator workflow tooling (consignment records, threshold tracking, dossier assembly, Registry-format export). Outside the scope of this paper.
- **kaimeter-app** — command-line and local HTTP interface; offline bundle import; proof export. Outside the scope of this paper.

Planned releases.

Release	Target	Content
v0.1	October 2026	kaimeter-rules : period, sector-dependent mark-ups, default-value selection, Art. 14 precursor averaging, scope table for current Annex I aluminium codes; sectors/aluminium/ primary route with slope-method PFCs; the test vector of Appendix B passing at full precision; bundle-hash binary; structure and stubs for all other sectors. No prover.

Release	Target	Content
v0.2	November 2026	First fixed-interpretter zkVM guest, evaluating the aluminium bundle as witness with in-circuit h_B ; differential tests between native and guest execution; overvoltage method; secondary aluminium; two-supplier precursor vector reproducing §11.
v0.3	Q1 2027	sectors/steel/ ; proof envelope and offline verifier; stable interpreter image IDs; attestation-policy schema and key resolution; signed bundle distribution; reproducible-build pipeline with SBOM and provenance.
Later	2027–2028	Cement and fertilisers (indirect emissions); hydrogen; Article 9 once adopted; downstream module as adopted; recursive proof composition; independent circuit/guest audit.

Everything in this paper described as planned or intended is not yet implemented. Readers are invited to check the repository’s status table, which will be kept consistent with this paper, and to report discrepancies.

11. Worked scenario: smelter $\rightarrow$ extruder $\rightarrow$ manufacturer $\rightarrow$ EU declaration

Actors. S , a primary aluminium smelter in China. S' , a second smelter. X , an extruder in China buying billet from S and S' . M , a manufacturer of aluminium window and door hardware in China, buying profiles from X — a product category of the kind proposed for inclusion from 2028. D , an EU importer (authorised CBAM declarant). V , an accredited verifier engaged by S , X and M .

Step 1 — Rules. All parties use bundle 2026.2.0 (hash h_B) for 2026 goods, incorporating IR 2025/2547 and the corrected default values of IR 2026/1740.

Step 2 — Smelters. S runs the bundle on 2026 activity data (as in §4.5) and obtains $y_S = 1.835$ (1.835038 at full precision). V performs verification, including the site visit, and signs an attestation over the verified inputs and h_B . S generates π_S . S' does likewise, obtaining $y_{S'} = 1.910$ and $\pi_{S'}$. Neither smelter’s activity data leaves its premises.

Step 3 — Extruder. X consumed 600 t of S 's billet and 430 t of S' 's billet to produce 1 000 t of profiles (CN 7604 10 90). Its own direct emissions — gas-fired homogenising and ageing furnaces — were 120 t CO₂. The bundle computes the weighted-average precursor SEE per Art. 14(2):

$$\bar{y}_{\text{prec}} = \frac{600 \times 1.835 + 430 \times 1.910}{1030} = 1.866$$

$$y_X = \frac{120}{1000} + 1.03 \times 1.866 = 0.120 + 1.922 = 2.042 \text{ t CO}_2\text{e} / \text{t}$$

X generates π_X , which verifies π_S and $\pi_{S'}$ internally. X learns y_S and $y_{S'}$ and nothing else about its suppliers' data — it already knows who they are, and the verification report for the profiles, a complex good, identifies both installations as §7.4 requires.

Step 4 — Declarant, 2026 goods. D imports 2 400 t of profiles from X in 2026. It receives π_X and y_X , verifies the proof in its own software, confirms h_B is an accepted bundle, and records total embedded emissions of $2\,400 \times 2.042 = 4\,901$ t CO₂e for these goods. Had D been unable to obtain actual values, it would have applied the China default for CN 7604 10 90 with the 2026 mark-up of 10 % — a figure the bundle also computes, so D can see the difference the proof made.

Step 5 — Certificates. By 30 September 2027, D 's declaration includes y_X and the quantity, the free-allocation adjustment per Art. 31 using the benchmark for the relevant route from IR 2025/2620, and — once the Article 9 implementing act is in force — any deduction for carbon price effectively paid under China's national ETS, for which S and X can generate a separate proof over allowance records without disclosing them. D surrenders the resulting certificates and retains π_X , the bundle hash, the accepted attestation-policy hash, the attested-input root and the verification report it is required to file and keep as evidence.

Step 6 — A rule change. Suppose in 2027 the Commission revises default values or a GWP. Bundle 2027.1.0 is published and signed. S , S' and X import it (offline, if necessary), re-run on the *same* private inputs, and issue new proofs. D holds both generations of proof and can show any reviewer exactly what changed and why.

Step 7 — The fourth tier, 2028. Suppose the downstream extension is adopted and M 's hardware enters scope from 1 January 2028. Bundle 2027.2.0 adds the relevant CN codes to `scope.rs` with `appliesFrom: 2028-01-01`. M buys 1 000 t of profiles from X (now carrying π'_X under the current bundle, $y'_X = 2.03$), machines and finishes them into 950 t of hardware with 50 t of pre-consumer scrap returned to X , and consumes 40 t CO₂ in its own finishing processes. V attests M 's bill of materials and process data. The downstream rule in `content.rs` computes:

$$y_M = \frac{40}{950} + \frac{950 \times 2.03}{950} = 0.042 + 2.030 = 2.072 \text{ t CO}_2\text{e} / \text{t}$$

(Treatment of the scrap flow and of any content coefficients follows whatever the adopted text specifies; the figure is illustrative.) M generates π_M , which verifies π'_X internally — and, through it, the smelters' proofs — without the proof disclosing to M which smelters supplied X , or to X M 's customers. The verification report for M 's complex good must nonetheless identify the installations at which its precursors were produced, and M 's verifier will have that information; identity is a property of the report, not of the proof (§7.4). D , or a different declarant importing hardware, verifies π_M and declares y_M . Absent the proof, the importer would default with the 30 % mark-up — on a product whose emissions are 98 % someone else's.

The chain has grown by one link. Nothing else has changed.

12. Related work and statement of novelty

Rules as code. OpenFisca (France; tax-benefit simulation), Catala (Merigoux, Chataing & Protzenko, 2021), Blawx (Morris) and the OECD OPSI *Cracking the Code* report (2020) established methods and governance for executable legislation. None address emissions methodology or bind encoded rules to cryptographic proofs of their execution.

Zero-knowledge proofs for sustainability claims. The closest prior work already composes proofs across a supply chain: Man et al. (2025, *Emission Impossible*) build privacy-preserving emissions claims across a chain of energy suppliers, data centres, cloud providers and customers; Babel et al. (2022) trace product-level emissions across tiers with shielded tokens while keeping business data confidential; Heiss et al. (2024) verify carbon accounting across supply chains; and smart-contract systems such as Tuli & Kim (2026) track and trade carbon footprints with zero-knowledge proofs. We therefore do **not** claim multi-tier composition as the delta. What these systems lack is rule identity: the computation is fixed by the application or circuit, not bound to a versioned, hash-identified encoding of a specific legal methodology. They cannot answer *which rule version* produced a number (R3), do not adapt deterministically as rules or scope change (R8), do not map onto the CBAM legal roles — accredited verifier, declarant, Registry (R4) — and do not address deployment under data-sovereignty constraints (R6).

Verified declarations as the incumbent. The established mechanism for publishing a verified figure without releasing plant data is the Environmental Product Declaration, independently verified under ISO 14025 and, for construction products, EN 15804; organisational greenhouse-gas statements are verified under ISO 14064-3 against ISO 14064-1. These already deliver much of R2 and R4, and they are the baseline this paper measures against rather than a null case. Their limitation is structural: verification attaches to a published declaration whose underlying methodology is not a versioned executable artifact, and there is no machine-checkable binding between the declared figure, the rule version applied and the verifier’s attestation.

Carbon data exchange. WBCSD PACT and Catena-X define exchange formats for footprint data and rely on organisational trust; the CBAM Operators Portal lets a non-EU operator register and share its verified embedded emissions with declarants, but registration and custody are central (in the Commission-operated Registry) and per installation, and the number carries no rule version.

Verifiable computation over regulation. Proposals for proof-carrying tax or KYC compliance exist in the cryptography literature, but to our knowledge none has combined a legally annotated, versioned, multi-sector rule bundle with recursive proofs along a physical supply chain for a regime with live financial liability.

Novelty claimed. The contribution is the *combination* and its specific engineering, and the delta differs by baseline. Against the incumbent Operators Portal, the residual value is threefold: (i) **no central custodian** — the operator need not register its installation or its verified figure in an EU-operated registry, and the proof travels directly to the declarant (R6); (ii) **composition across tiers** — a downstream good’s embedded emissions are assembled from precursor values held by independent operators, which a per-installation portal does not do (R5); and (iii) **rule identity** — the figure is bound to a hash-identified rule version, so it can be re-verified as rules and scope change (R3, R8). Against the zero-knowledge literature, which already composes proofs across supply tiers, the delta is rule identity and legal-role mapping — a trust-anchor model that places accredited verification, the declarant’s liability and the Registry, not the proof, at the root (R4). We make no claim of novelty for zero-knowledge proofs, rules as code, or recursive proof composition individually.

13. Limitations and open questions

We prefer to state these ourselves.

1. **Garbage in, proven out.** A proof cannot detect falsified inputs. The architecture’s integrity rests on the attestation layer, and ultimately on accredited verification. Self-attestation is excluded structurally — attester keys resolve from an accepted policy to accreditation identifiers (§5.1) — but the policy’s completeness and the accreditation process remain the root of trust. Where verification is unavailable, instrument attestations are weaker evidence and should be described as such.
2. **Verifier capacity.** The bottleneck the architecture is designed to relieve is also the anchor it depends on. It reduces the *marginal* cost of using one verification many times; it does not create verifiers.
3. **Implementation status.** At the time of this preprint the reference implementation has not been released. The claim that the architecture is sector-agnostic rests on the bundle structure and on the observation that each sector’s methodology is arithmetic over attested inputs in the same way; it has not yet been demonstrated in code for any sector. Cement and fertilisers, with indirect emissions, will be the first real test of `common/indirect.rs`.
4. **Downstream methodology is not yet law.** §6.4 and Step 7 of §11 describe a proposed regime. The product list, the content model, the treatment of scrap and the anti-circumvention rules may all change in trilogue. The `downstream/` module should be regarded as a design placeholder until the text is adopted.
5. **Confidentiality is narrower than “no disclosure”.** The architecture withholds activity data, not the statutory compliance record. In the case of complex goods, Annex VI, point 2(k)(iii) requires the verification report to identify the installations that produced the precursors; Article 6(2)(d) places copies of the report in the declaration; and Articles 7(5)–(6) with Annex V, point 2 require the declarant to keep the installation identification, the report, the specific embedded emissions and the method used until the end of the fourth year after the declaration is submitted. The proof does not carry those items, but nothing here suppresses them, and the confidentiality claim must be read at the scope of §7.4. The legal fit in §7 has not been reviewed by counsel.
6. **Provable-evaluator correctness.** No prover exists yet. When it does — as the fixed interpreter of §5.4 or a hand-written circuit — the differential test suite will be strong evidence but not proof of equivalence with the plain evaluator, and a soundness bug would be a silent failure. The in-circuit canonical-hash implementation is itself part of what must be audited. Independent audit is a prerequisite for anyone relying on proofs for financial decisions.
7. **Regulatory acceptance.** No competent authority has stated how it will treat proofs as evidence. Until one does, the proof’s practical value is as *superior internal evidence*, not as a recognised compliance instrument.
8. **Article 9.** Rules for recognising third-country carbon prices remain pending. Proofs over carbon-price deductions should not be issued until the implementing act is adopted.
9. **Fixed-point and rounding.** Mismatches between the bundle’s rounding and the Registry’s internal computation could produce small discrepancies. We will track this in the test vectors and welcome reference examples from the Commission.
10. **Identity and key management.** Attestations require verifiers and operators to hold signing keys. Key compromise, rotation and revocation rely for now on conventional PKI practice.

11. **Governance of the bundle.** Whoever maintains the canonical bundle holds influence over what “the rules as code” say — and, as scope expands, over which goods are in it. §14 addresses this; we regard neutral, multi-stakeholder governance as necessary for legitimacy.
 12. **Export controls and sanctions.** The software includes cryptographic components and is intended for use in China. Publicly available source code is treated favourably under US export regulations, but installation and support services remain subject to restricted-party rules. This is a compliance matter for any service provider, not a property of the software.
-

14. Open-source governance and sustainability

The reference implementation is released under the Apache License 2.0, chosen for its explicit patent grant and its acceptance by enterprise and public-sector legal teams in the EU, China and the United States. Contributions are accepted under the Developer Certificate of Origin. Kaimeter is a trademark of Keldrion, LLC; the trademark is reserved, and forks are free to use the code but must use their own name.

The rule bundles are published under the same licence. We expect and encourage independent implementations of the evaluator against the published test vectors; conformance, not code, is the standard. We particularly encourage sector bodies — steel, cement and fertiliser associations, and their verifiers — to contribute and review the modules for their sectors, since their members will bear the consequences of an error.

We intend to seek a neutral foundation home for the project so that no single company — including the author’s — controls the canonical bundle. Until then, the maintainers’ signing keys and change process will be published, every change will carry its legal motivation, and the changelog will be public.

The company that publishes this paper earns revenue from training, deployment, integration, support and delegated filing services around the software. It does not earn revenue from the software itself, and the software is designed to outlive any single provider: an operator’s proofs, bundles and evidence remain verifiable by anyone with the open-source verifier regardless of whether the original provider continues to exist.

15. Roadmap and generalisation

Near term (to the first declaration cycle, September 2027). Release v0.1 (October 2026) and v0.2 (November 2026) as set out in §10; complete precursor recursion for all aluminium production routes; implement **sectors/steel/** — the largest CBAM import volume and the deepest precursor chains; publish conformance test vectors matched to Commission guidance examples; achieve reproducible builds with published provenance; commission an independent audit of the provable evaluator; pilot with a small number of operator–declarant pairs; engage competent authorities and at least one accredited verifier on the attestation format.

Medium term (2027–2028). Implement cement and fertilisers, exercising the indirect-emissions path; hydrogen; encode the Article 9 deduction once its implementing act is adopted; encode the downstream extension as adopted, with the product list as data and the content

model as rule, in time for operators to test a full year before 1 January 2028; propose the proof envelope to a standards body.

Generalisation. The architecture is jurisdiction-agnostic by construction: a rule bundle is a jurisdiction, a set of sectors and a period. The UK CBAM (from January 2027) and prospective US measures such as the proposed Foreign Pollution Fee Act — which, as introduced, would cover steel, aluminium and other materials — would each be a bundle over the *same* attested installation data. An operator verified once could prove compliance to three regimes without three disclosures. We regard “one attested dataset, many jurisdictions’ rules, many proofs” as the long-term value of separating the rules from the data — and the annual expansion of CBAM’s own scope as the first demonstration that the separation is necessary.

Acknowledgements

This preprint has not yet been externally reviewed. Comments are welcome at the address on the title page and will be acknowledged in a subsequent version. Errors are the author’s.

Disclaimer

This paper describes a technical architecture. It is not legal advice, and it does not purport to state how any competent authority will treat the evidence it describes. Statements about the proposed downstream extension describe legislative proposals that had not been adopted at the time of writing. Statements about the reference implementation describe planned work. Readers should rely on the cited legal instruments and on qualified counsel.

References

Legal instruments (via EUR-Lex, <https://eur-lex.europa.eu>; see Appendix C for roles):

1. Regulation (EU) 2023/956 of 10 May 2023 establishing a carbon border adjustment mechanism.
2. Regulation (EU) 2025/2083 of 8 October 2025 amending Regulation (EU) 2023/956 as regards simplifying and strengthening the carbon border adjustment mechanism. ELI: <http://data.europa.eu/eli/reg/2025/2083/oj>
3. Commission Implementing Regulation (EU) 2025/2547 of 10 December 2025 on the methods for the calculation of emissions embedded in goods. ELI: http://data.europa.eu/eli/reg_impl/2025/2547/oj
4. Commission Implementing Regulation (EU) 2025/2550 amending Implementing Regulation (EU) 2024/3210 as regards the CBAM registry.
5. Commission Delegated Regulation (EU) 2025/2551 of 20 November 2025 on accreditation of verifiers.
6. Commission Implementing Regulation (EU) 2025/2620 (CBAM benchmarks).
7. Commission Implementing Regulation (EU) 2025/2621 (default values), as corrected by Commission Implementing Regulation (EU) 2026/1740 of 20 July 2026. ELI: http://data.europa.eu/eli/reg_impl/2026/1740/oj

8. Commission Implementing Regulation (EU) 2018/2066 (Monitoring and Reporting Regulation).
9. European Commission, *Proposal for a Regulation amending Regulation (EU) 2023/956 as regards extending the scope to certain downstream products and strengthening anti-circumvention*, COM(2025) 989, 17 December 2025.
10. Council of the European Union, general approach on COM(2025) 989, 12 June 2026.
11. European Parliament, ENVI Committee report on COM(2025) 989 (rapporteur M. Chahim), adopted 6 July 2026.
12. European Commission, DG TAXUD, *Guidance Document 3: CBAM methods for the calculation of emissions embedded in goods*, August 2026.

Literature:

13. OECD Observatory of Public Sector Innovation, *Cracking the Code: Rulemaking for Humans and Machines*, 2020.
14. Merigoux, D., Chataing, N., Protzenko, J., “Catala: A Programming Language for the Law”, *Proc. ACM Program. Lang.* (ICFP), 2021.
15. OpenFisca, <https://openfisca.org>.
16. Morris, J. *Blawx: A user-friendly web-based tool for Rules as Code*. GitHub repository: <https://github.com/Lexpedite/blawx>.
17. Man, J., Jaffer, S., Ferris, P., Kleppmann, M., & Madhavapeddy, A. (2025). Emission Impossible: privacy-preserving carbon emissions claims. arXiv preprint arXiv:2506.16347. <https://doi.org/10.48550/arXiv.2506.16347>
18. WBCSD, *PACT Technical Specifications for PCF Data Exchange*.
19. Catena-X Automotive Network, *PCF Rulebook*.
20. Heiss, J., Oegel, T., Shakeri, M., & Tai, S. (2024). Verifiable Carbon Accounting in Supply Chains. *IEEE Transactions on Services Computing*, 17(4), 1861–1874. <https://doi.org/10.1109/TSC.2023.3332831>
21. Babel, M., Gramlich, V., Körner, M.-F., Sedlmeir, J., Strüker, J., & Zwede, T. (2022). Enabling end-to-end digital carbon emission tracing with shielded NFTs. *Energy Informatics*, 5(S1), art. 27. <https://doi.org/10.1186/s42162-022-00199-3>
22. Tuli, E. A., & Kim, D.-S. (2026). Individual CF tracking and management system using blockchain and zero-knowledge proof. *ICT Express*, 12(3), 646–652. <https://doi.org/10.1016/j.ict.2026.03.008>
23. International Organization for Standardization, *ISO 14064-1:2018 — Greenhouse gases — Part 1: Specification with guidance at the organization level for quantification and reporting of greenhouse gas emissions and removals*.
24. International Organization for Standardization, *ISO 14064-3:2019 — Greenhouse gases — Part 3: Specification with guidance for the verification and validation of greenhouse gas statements*.
25. International Organization for Standardization, *ISO 14025 — Environmental labels and declarations — Type III environmental declarations — Principles and procedures* (2006; revised 2026).
26. European Committee for Standardization, *EN 15804:2012+A2:2019 — Sustainability of construction works — Environmental product declarations — Core rules for the product category of construction products*.
27. Goldwasser, S., Micali, S., Rackoff, C., “The Knowledge Complexity of Interactive Proof Systems”, *SIAM J. Comput.*, 1989.
28. Ben-Sasson, E., Chiesa, A., Tromer, E., Virza, M., “Scalable Zero Knowledge via Cycles of Elliptic Curves”, CRYPTO 2014.
29. Reproducible Builds project, <https://reproducible-builds.org>; SLSA, <https://slsa.dev>.

Appendix A — Glossary

Term	Meaning
Authorised CBAM declarant Operator	The EU-established importer or indirect customs representative legally responsible for declaring and surrendering certificates
Accredited verifier	The person operating a non-EU installation producing CBAM goods, at any tier
Complex good / precursor	A body accredited under Delegated Regulation 2025/2551 to verify embedded emissions
Downstream good	A good whose production consumes other CBAM goods (precursors), whose embedded emissions must be included
Annex II sector	A finished or semi-finished product proposed for inclusion from 2028 whose embedded emissions derive mainly from its CBAM-material content
SEE	A sector for which only direct emissions count: iron and steel, aluminium, hydrogen and, since Regulation (EU) 2025/2083, electricity
Default value	Specific embedded emissions, t CO ₂ e per tonne of good
Rule bundle	Country- and CN-code-specific fallback emissions figure, with a sector-dependent mark-up
Attestation	A versioned, hash-identified set of executable rules and parameter tables encoding a methodology across one or more sectors
Zero-knowledge proof	A signed statement about inputs (from a verifier, an instrument, or an upstream proof)
Recursive proof	A cryptographic proof that a statement is true which reveals nothing beyond the truth of the statement
zkVM	A proof that verifies one or more other proofs as part of its own statement
Air-gapped	A zero-knowledge virtual machine: a proof system that proves the correct execution of an ordinary compiled program
	Operated on a host with no network connectivity

Appendix B — Test vector for the worked example

The following vector fixes the worked example of §4.5 at the bundle’s fixed-point precision. It is the first conformance vector for `kaimeter-rules` and will be included, unchanged, in the v0.1 release.

```
{  
  "bundle": "cbam",  
  "sector": "aluminium",  
  "route": "primary",  
  "cn_code": "7601",  
  "period": "2026",  
}
```

```

"method": "pfc_slope",
"parameters": {
  "gwp_cf4": "6630",
  "gwp_c2f6": "11100"
},
"inputs": {
  "production_t": "100000",
  "direct_co2_t": "155000",
  "anode_effect_minutes_per_cell_day": "0.25",
  "slope_emission_factor_cf4": "0.143",
  "weight_fraction_c2f6_cf4": "0.121"
},
"expected": {
  "e_cf4_t": "3.575000",
  "e_c2f6_t": "0.432575",
  "e_pfc_tco2e": "28503.832500",
  "see_tco2e_per_t": "1.835038"
}
}

```

All numeric values are decimal strings; implementations must not parse them as binary floating point.

Appendix C — Legal instruments and their roles

Instrument	Role
Regulation (EU) 2023/956	Basic CBAM Regulation; Annex I goods; Annex II direct-only sectors; Annex IV calculation methods
Regulation (EU) 2025/2083	Simplification amendments: 50-tonne threshold, 30 September deadline, certificate sales from 1 Feb 2027, 50 % holding rule, delegated submission
IR (EU) 2025/2547	Methods for calculating embedded emissions in the definitive period
IR (EU) 2025/2620	Benchmarks for the Art. 31 free-allocation adjustment
IR (EU) 2025/2621, corrected by IR (EU) 2026/1740	Default values and sector-dependent mark-ups (10/20/30 % for most sectors; 1 % for fertilisers)
IR (EU) 2024/3210, as amended by IR (EU) 2025/2550	CBAM Registry and Operators Portal: operator registration, the operator's portal as unique entry point, and operator-verifier-declarant data exchange
DR (EU) 2025/2551	Accreditation of verifiers
COM(2025) 989 and legislative positions	Proposed downstream extension, scrap, anti-circumvention, annual review (not adopted at time of writing)