Testing Machine-Readable Compliance: Validation Gaps and Omission Invariance in Stablecoin Reporting

Imane El Imami[1,2], Jeonghoon Oh[1], Jason Meyers[3], Gerard Rod Brennan[1], Miklos Vasarhelyi[1], Alexander Sannella[1], Thomas Egan[3], Said Ouatik El Alaoui[2]

[1]*Rutgers Continuous Auditing and Reporting Lab (CARLab), Rutgers Business School, Rutgers University, New Jersey, United States*
[2]*Engineering Sciences Laboratory, National School of Applied Sciences, Ibn Tofail University, Kenitra, Morocco*
[3]*Auditchain Labs AG*

imane.elimami@uit.ac.ma
ie138@scarletmail.rutgers.edu
jo610@scarletmail.rutgers.edu
jm@auditchain.com

Abstract

Machine-readable regulatory reporting is often implemented on the assumption that structured disclosure also delivers automated enforcement. We examine that assumption in the Blockchain Network Participation (BNP) extension to the Federal Deposit Insurance Corporation's GENIUS Act proposed reporting forms, including weekly PS-01, reduced-burden PS-01a, and quarterly PS-02. Using a four-layer evidence pipeline from the official forms to the current JSON source models, generated XBRL packages, and processor-oriented XBRL Formula analysis, we audit the current validation corpus and reconcile source models with delivered artifacts. Structural content survives generation exactly, but the validation layer is transformed: some assertions are synthesized rather than preserved as authored, severity and diagnostic messages are absent, and static analysis predicts substantial zero-binding exposure in the reduced-burden package. A malformed quarterly assertion that previously aborted Formula processing is absent from the current artifact. We further establish an omission-invariance proposition: when all completeness-relevant operands originate within the filing, a consistently omitted affiliate can leave internal validation satisfied. Perimeter and attestation controls strengthen accountability but cannot independently detect the missing entity. The study therefore distinguishes internal consistency from reporting completeness and shows why external anchoring is necessary when validation depends on self-reported reporting boundaries.

Keywords: XBRL Formula, Value Assertion, Validation Rules, Continuous Auditing, Regulatory Data Quality, Disclosure Completeness, Stablecoins, GENIUS Act, Legal Entity Identifier, Typed Dimensions, Anti-Evasion

1 Introduction

A disclosure taxonomy gives a reporting entity a vocabulary. A validation rule set determines which statements in that vocabulary the regulator has undertaken to check. The distinction is

easy to elide, because both are delivered in the same technical wrapper and because a mandate for "machine-readable" data is naturally, but wrongly, heard as a mandate for machine-*checked* data. Fifteen years of evidence from the Securities and Exchange Commission's XBRL program show what the elision costs: filings that are syntactically valid and substantively unreliable, with error rates that structure alone did not remove (Debreceny et al. 2010; Du et al. 2013). Whether the new generation of machine-readable *prudential* regimes repeats that history depends on a layer that has received limited independent scrutiny, namely the executable rules that stand between a well-formed filing and a supervisable one.

This paper examines that validation layer in the emerging GENIUS Act reporting environment. The Guiding and Establishing National Innovation for U.S. Stablecoins Act (GENIUS Act, Pub. L. No. 119-27, codified at 12 U.S.C. §§5901–5919)[1] requires permitted payment stablecoin issuers (PPSIs) to file recurring activity, reserve, and condition reports, and the Federal Deposit Insurance Corporation (FDIC) has published the corresponding proposed forms: the weekly PS-01, a reduced-burden weekly PS-01a for smaller issuers, and the quarterly PS-02.[2] A companion study (El Imami et al. 2026) argued that these reserve-centric forms leave the blockchain infrastructure on which redemption actually depends unobserved, a structural misconception termed the "Illusion of the Vault," and proposed a 148-question, 125-element Blockchain Network Participation (BNP) disclosure taxonomy as the instrument through which infrastructure risk becomes supervisable. That paper made a design-and-necessity argument. It asserted, but could not test, the property on which its supervisory value rests: that the architecture renders any omission programmatically identifiable without requiring a regulator to intervene.

The present paper tests that property. We ask whether the machine-readable models that implement the FDIC forms and the BNP extension actually deliver, in the executable artifacts processed by a validation system, the enforcement of the taxonomy that is claimed to embody. The question decomposes into a fidelity problem and a logic problem. The fidelity problem is empirical: a source model must survive compilation into the XBRL package that a validation system will actually load, and there is no *a priori* reason to expect that a correct source produces a correct executable. The logic problem is theoretical: even a perfectly compiled rule set can only test relationships among facts the filer chose to report, and the taxonomy's central anti-evasion claim concerns precisely the facts a filer might choose not to report.

Four research questions organize the study:

- RQ1: Does the current validation architecture implement the three-tier (structural, arithmetic, and temporal) enforcement design asserted in the regulatory submissions, and how does the reduced-burden form differ from the standard form?
- RQ2: Does the source model survive generation into the delivered XBRL package, with respect to concepts, dimensions, facts, and, critically, the executable assertions themselves?
- RQ3: What deficiencies are revealed by standards-conformant processor-oriented evaluation, and which deficiencies are current rather than already remediated?
- RQ4: Can the current rule corpus detect the consistent omission of a reportable affiliate, and what external anchor would make such omissions detectable?

[1] GENIUS Act, Pub. L. No. 119-27, 12 U.S.C. §§5901–5919 (2025). https://www.govinfo.gov/app/details/PLAW-119publ27

[2] FDIC Reporting Forms PS-01 (FDIC 6930/01), PS-01a (FDIC 6930/01A), and PS-02 (FDIC 6930/02), OMB Number 3064-0225, filed under 12 U.S.C. §5905 through FDICconnect.

Our answers rest on a strict evidence hierarchy: the current source models, the packages generated from them, the official FDIC forms, and, only where explicitly labeled, runtime evidence from earlier model generations that our forensic audit confirmed and that the current generation has since remediated. Every count reported here was computed directly from the shipped artifacts rather than transcribed from project documentation.

The results are double-edged, and the paper is organized to keep the edges apart. On one side, the architecture is substantially real. The three current models carry 913 validation rule rows, of which 126 implement the BNP extension across nine functional families; the structural and arithmetic tiers are implemented in full; and the structural content of the models, comprising 1,282, 153, and 771 facts together with every presentation, calculation, and dimensional relationship, survives into the generated packages exactly. A defect that once silently aborted formula processing for every quarterly filing has been fixed, and the fix is visible in the current artifact. On the other side, the validation layer degrades in generation in ways invisible to anyone who does not execute it: assertions synthesized over raw names instead of variables, a complete absence of severity and messages, and a reduced-burden package whose shipped instance leaves an estimated three-fifths of its assertions with nothing to evaluate. Beneath both edges sits the logical result. Every completeness-relevant assertion in the corpus draws every operand from the filing itself, so a consistently omitted affiliate fails nothing. The attestation machinery in the current models converts that omission from a silent gap into an affirmative misstatement by a named officer, which is a legal repair rather than a computational one. Detection requires an operand outside the filer's control, and we specify where such an operand exists.

The contribution is fourfold. Theoretically, we state and prove an omission-invariance proposition that generalizes beyond stablecoins to every regime that polices its reporting boundary through internal consistency, including consolidated-group reporting, related-party disclosure, beneficial ownership, and sustainability boundaries. Methodologically, we develop an evaluation protocol for platform-generated XBRL Formula built on a presence–execution ladder, under which an assertion may exist, parse, execute, bind its intended facts, and pass, with each rung requiring separate evidence, together with a mechanical signature that separates authored from synthesized assertions. Empirically, we provide an independently computed census and source-to-package reconciliation of the current GENIUS Act validation corpus, including the reduced-burden form, which was not included in our earlier validation analysis. For regulatory practice and continuous auditing, we translate the findings into a ranked set of pipeline controls, namely severity, delivered-artifact gating, and external anchoring, that separate what automation can already accomplish from what only an operand outside the filing can accomplish.

The remainder proceeds as follows. Section 2 frames the study in the literature it requires. Section 3 sets out the artifacts and the evidence pipeline. Section 4 reports results by research question, distinguishing among current results, historical defects now fixed, and proposed future controls. Section 5 discusses the implications, Section 6 states the limitations, and Section 7 concludes.

2 Literature and Theoretical Framing

2.1 Machine-Readable Regulatory Data and the Quality Gap

The XBRL literature has long established that mandated structure does not produce reliable data. Debreceny et al. (2010) documented pervasive calculation inconsistencies in initial filings

to the SEC; Du et al. (2013) counted filing errors in the thousands and asked, pointedly, what followed from them; Debreceny et al. (2011) showed that preparer-controlled extensions undermined comparability; and Bartley et al. (2011) found material discrepancies between XBRL filings and the official documents they tagged. Subsequent work demonstrated that the data nonetheless changed the information environment, reducing information asymmetry and processing costs for first-wave filers (Kim et al. 2012; Blankespoor et al. 2014), and that it became research infrastructure in its own right (Hoitash and Hoitash 2018). The arc of this literature is the premise of our study: machine-readability changed what could be *consumed* long before anyone verified what was being *enforced*. The enforcement layer, comprising XBRL Formula assertions, dimensional constraints, and the pipelines that execute them, has been specified in open standards[3] but seldom independently audited in a live regulatory corpus. Boritz and No (2009) provided an early demonstration that assurance over XBRL artifacts requires procedures distinct from assurance over the underlying report; we extend that insight from instance documents to the validation rules themselves.

2.2 Continuous Auditing and Automated Validation

Continuous auditing theory has long held that assurance should move to the cadence of the underlying process (Vasarhelyi and Halper 1991), with implementations demonstrating automated, rule-based monitoring at scale (Alles et al. 2006; Kogan et al. 2014) and a broadening analytic apparatus (Appelbaum et al. 2017). Stablecoin infrastructure is the limiting case for this programme: blocks settle in seconds, extractable-value opportunities appear and vanish within a single block (Daian et al. 2020; Qin et al. 2022), and bridge exploits complete before any periodic attestation cycle begins (Belenkov et al. 2025; Notland et al. 2026). The companion paper develops why reserve-centric supervision under-observes these risks (El Imami et al. 2026), consistent with broader financial-stability evidence that redemption-rail integrity, and not reserve quality alone, determines peg survival (Diop et al. 2024; Eichengreen et al. 2025). We take from this literature only what the present study requires. If supervision is to operate at machine cadence, the validation layer is the instrument of supervision, and its own quality is therefore an assurance object, one this paper treats with the discipline continuous auditing applies to transaction streams, including the principle that a monitoring system must be tested for its ability to report failure and not only success (Kogan et al. 2014).

2.3 Internal Consistency versus Completeness

The classical unravelling results hold that when disclosure is costless and verifiable, and the audience knows the sender is informed, non-disclosure is interpreted adversely and full disclosure emerges (Grossman 1981; Milgrom 1981). The equilibrium weakens when the audience cannot determine whether the sender is informed (Dye 1985), when disclosure is costly (Verrecchia 1983), or when the audience cannot verify what has been withheld (Beyer et al. 2010). The latter condition is the one operationalized in this study. A validation rule set is, in these terms, a verification technology. Proposition 1 below locates precisely what that technology can verify: relationships *within* the disclosed set, and never the boundary *of* the disclosed set. The attestation construction in the current models removes the Dye condition, since the filer affirmatively asserts that it is informed and that its enumeration is complete, which converts omission into misstatement and creates enforcement value. Restoring unravelling, however, requires the receiver to enumerate or bound what was not disclosed, and that requires an operand the sender does not control. This reframing connects a specification-level property

[3] XBRL International, *XBRL Dimensions 1.0* (2006) and *XBRL Formula 1.0* (2009), Recommendations. https://specifications.xbrl.org

of XBRL Formula to a long-standing theoretical literature and, we argue, generalizes to every self-described reporting perimeter.

2.4 Entity Anchoring: The LEI, Level 2, and the Verifiable LEI

The Global LEI System supplies the candidate's external operand. Level 1 identifies entities; Level 2 publishes each entity's direct and ultimate accounting-consolidating parents, defined by reference to accounting-consolidation standards and verified by issuing organizations against public documents.[4] Since the joint data standards adopted under the Financial Data Transparency Act designated the LEI for U.S. financial regulators, the identifier is native to the agencies that receive these filings.[5] The verifiable LEI under ISO 17442-3 extends the system from identification to cryptographically verifiable organizational credentials, permitting a role credential, that is an identified officer of an identified entity, to bind an attestation.[6] Neither construct appears as an executable operand in any current model. Section 4.5 develops them strictly as design propositions, together with the accounting-consolidation perimeter under ASC 810 and IFRS 10 as the upstream anchor whose exteriority that section formalizes.

3 Research Design

3.1 Setting and Artifacts

The unit of analysis is the machine-readable implementation of three official FDIC reporting forms for permitted payment stablecoin issuers: Form PS-01, the weekly activity and reserve report for issuers at or above $1 billion in outstanding issuance value or $100 million in average daily transaction volume; Form PS-01a, the reduced weekly report for smaller issuers; and Form PS-02, the quarterly report of condition and income. Each form is implemented as a source model in a structured JSON serialization carrying the concept dictionary (Terms), the presentation, definition, and calculation networks (Structures and Associations), a reference fact instance (Facts and Facts-Dimensions), and five rule collections: four declarative (Consistency, RollForward, MemberAggregation, and Derivation) and one executable (Nonstandard), whose rows embed complete XBRL Formula extended links. From each model the reporting platform generates a delivery package comprising a taxonomy schema, label, reference, presentation, calculation, and definition linkbases, a formula linkbase, an XBRL instance, and an inline-XBRL rendering. The models extend the official forms with the BNP schedules covering issuer network participation, affiliate participation, and aggregated concentration and systemic risk, implemented on two typed axes, `BlockchainNetworkIdentifierTypedAxis` and `AffiliateIdentifierTypedAxis`, that do not appear on the published forms.

Analysis observes a strict evidence hierarchy: first, the current source models; second, the packages generated from them together with reproducible runtime evidence; third, the official FDIC forms as ground truth for regulatory content; fourth, the companion paper for intellectual continuity; and fifth, earlier model generations only where a finding's remediation history is itself the evidence. No count from any prior document is reproduced without recomputation.

[4] LEI Regulatory Oversight Committee (2016), *Collecting Data on Direct and Ultimate Parents of Legal Entities in the Global LEI System – Phase 1*. https://www.leiroc.org

[5] Financial Data Transparency Act joint data standards, final rule, 91 Fed. Reg. 38,246. https://unblock.federalregister.gov/

[6] ISO 17442-1:2020, *Financial services — Legal entity identifier (LEI) — Part 1: Assignment*; ISO 17442-3, *Part 3: Verifiable LEIs (vLEIs)*. https://www.gleif.org/en/organizational-identity/lei-vlei/the-legal-entity-identifier-lei

3.2 The Evidence Pipeline

Figure 1 presents the four-layer pipeline and the analysis performed at each interface.

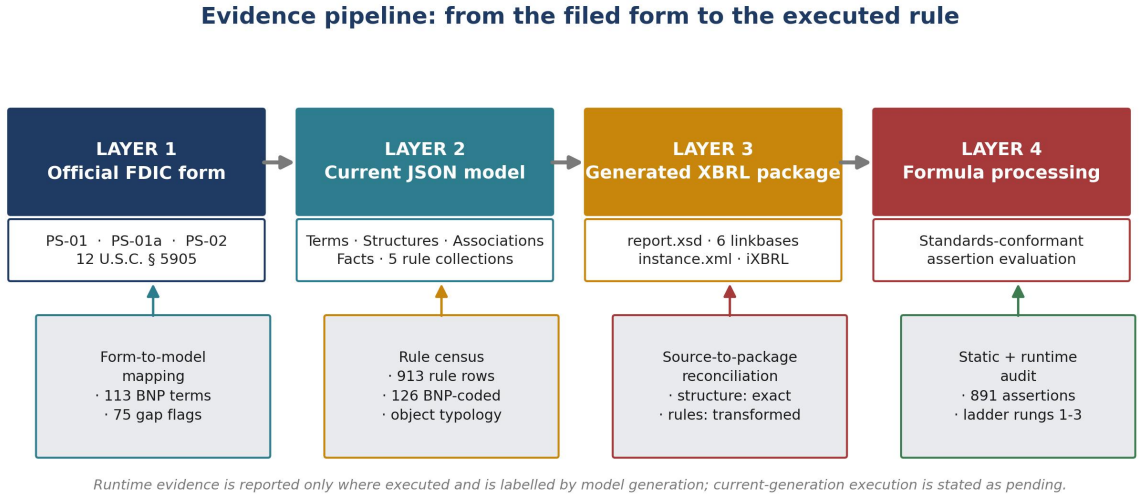

Figure 1: The four-layer evidence pipeline, from the official FDIC form to the executed rule

Layer 1 to Layer 2 (form-to-model mapping). We verified that each model implements its form's schedules (PS-01 Schedules A through H, the weekly-statistic restatement of PS-01a, and PS-02 Schedules A through E) and isolated the project's extensions: 113 BNP framework terms per model, comprising 99 concepts, four hypercubes, three line-item groups, five abstracts, and two typed dimensions; the BNP schedules; and 29, 29, and 17 elements that the models explicitly flag as awaiting regulator specification.

Layer 2 (model census). For each model we computed the population of every collection: terms by report-element category, structures, associations by network, facts and fact-dimension rows, every rule collection, and, within the executable collection, the embedded XBRL object types. BNP attribution uses the rule-code token, and the nine BNP families are read from the structured identifier scheme.

Layer 2 to Layer 3 (source-to-package reconciliation). For each package we parsed every generated file and reconciled element declarations against terms; hypercubes, dimensions, and typed-domain references; presentation, calculation, and definition arcs against association rows; instance facts and contexts against the model's fact and fact-dimension tables; references and labels; and the formula linkbase's assertions against the model's rule codes, resolving the generator's conventions of suffix-stripping and declarative compilation.

Layer 3 (formula corpus audit). For every assertion we extracted the identifier, test expression, variable set, filters, and parameters, and computed a battery of static integrity checks covering duplicate identifiers, references to undeclared concepts, unresolvable variable references, unbalanced delimiters, empty tests, and the presence of severity and message resources. Two diagnostic signatures were developed for this study: the *authored/synthesized signature*, which distinguishes tests written over declared variables from tests written over raw qualified names, and a *zero-binding estimate*, which flags assertions whose concept filters target concepts unpopulated in the shipped instance and which therefore cannot evaluate against it.

Layer 3 to Layer 4 (runtime evidence). We maintain a hard distinction, used

throughout and presented in Figure 2: an assertion may exist, parse, execute, bind its intended facts, and pass, and no rung of that ladder implies the next. Static analysis establishes the first two rungs and constrains the third; only a standards-conformant processor establishes the remainder. Runtime evidence in this paper comes from the project's earlier evaluation battery of eleven packages spanning two model families and four generations, executed with formula processing enabled, including negative controls in which three seeded linkbase defects each produced the expected diagnostic and a nine-defect mutation harness in which each seeded violation reddened only its intended rules. That battery is reported only where our forensic re-audit of the current artifacts confirms its relevance, and always labelled by generation. We report no runtime results for the current-generation packages; that execution is scheduled, and Section 4.4 states in advance the specific static predictions it will adjudicate.

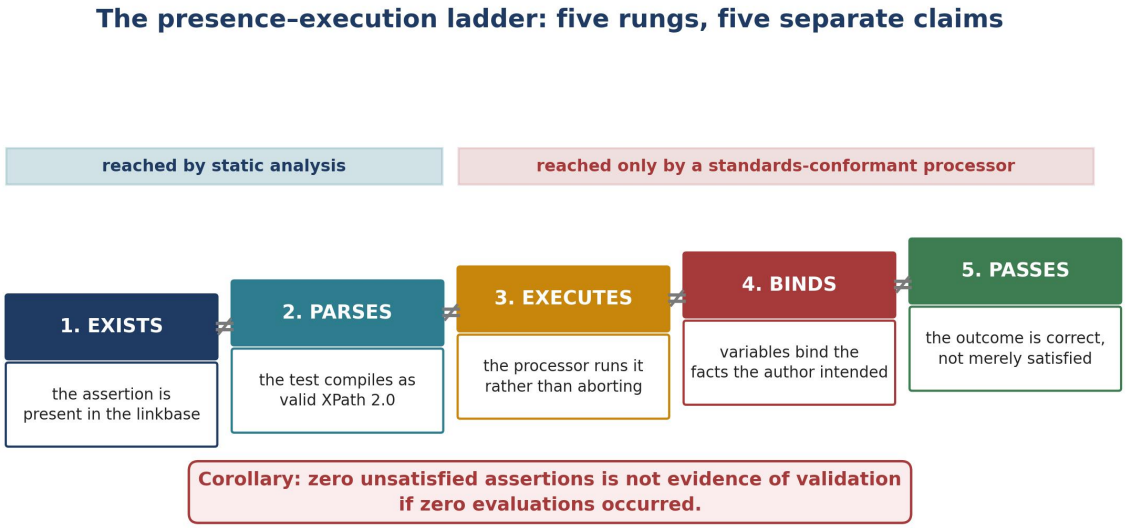

Figure 2: The presence–execution ladder: five rungs, five separate evidentiary claims

3.3 Reproducibility

All quantitative results reported in this study were derived from the three current JSON source models, their corresponding generated XBRL packages, and the official FDIC forms. Model-level counts were recomputed directly from the source artifacts rather than transcribed from prior project documentation. Source-to-package reconciliation was performed across concepts, dimensional structures, facts, associations, and executable assertions. Where the study relies on runtime evidence from earlier model generations, those results are explicitly identified as historical rather than current-generation evidence. The current JSON models and generated XBRL packages constitute the primary reproducibility artifacts for the study.

3.4 AI-Assisted Research Workflow and Human Oversight

Generative AI tools were used as research-support instruments for artifact exploration, cross-model comparison, and manuscript development. The AI-generated outputs were not treated as independent evidentiary sources. All quantitative findings reported in the study were verified against the underlying regulatory forms, current JSON models, and generated XBRL packages, and the authors reviewed and reconciled substantive interpretations. Final analytical judgments, interpretation of findings, and manuscript decisions remained under human author control. The use of generative AI did not modify the source models or the underlying research artifacts.

4 Results

4.1 RQ1: The Validation Architecture as Built

Table 1 reports the complete rule census of the three current models, and Figure 3 presents the same corpus graphically alongside the decomposition of the BNP layer into families and tiers. The corpus comprises 913 rule rows: 332 on PS-01, 329 on PS-01a, and 252 on PS-02. The executable channel dominates, with 303, 303, and 226 Nonstandard rows, each embedding a complete XBRL Formula extended link, against a deliberately thin declarative layer of 29, 26, and 26 rows retained for the platform's native evaluator. Within the executable channel the object typology is nearly uniform: 300, 300, and 225 value assertions; two existence assertions on each weekly model and none on the quarterly; and one parameter-register row per model defining 61, 61, and 59 named parameters that centralize every tolerance, bound, and threshold the assertions consume. There are no consistency assertions, no assertion sets, and no formula elements producing derived facts. There is also, as Section 4.4 develops, no severity classification and no authored validation message anywhere in the corpus.

Table 1: Validation rule census, current models (computed directly from source)

Rule collection	**PS-01**	**PS-01a**	**PS-02**
Rules-Consistency	6	6	9
Rules-RollForward	1	1	1
Rules-MemberAggregation	15	12	11
Rules-Derivation	7	7	5
Rules-Nonstandard (executable)	303	303	226
Total rule rows	**332**	**329**	**252**
of which BNP-coded	41	41	44
Embedded value assertions	300	300	225
Embedded existence assertions	2	2	0
Parameter registers (parameters defined)	1 (61)	1 (61)	1 (59)

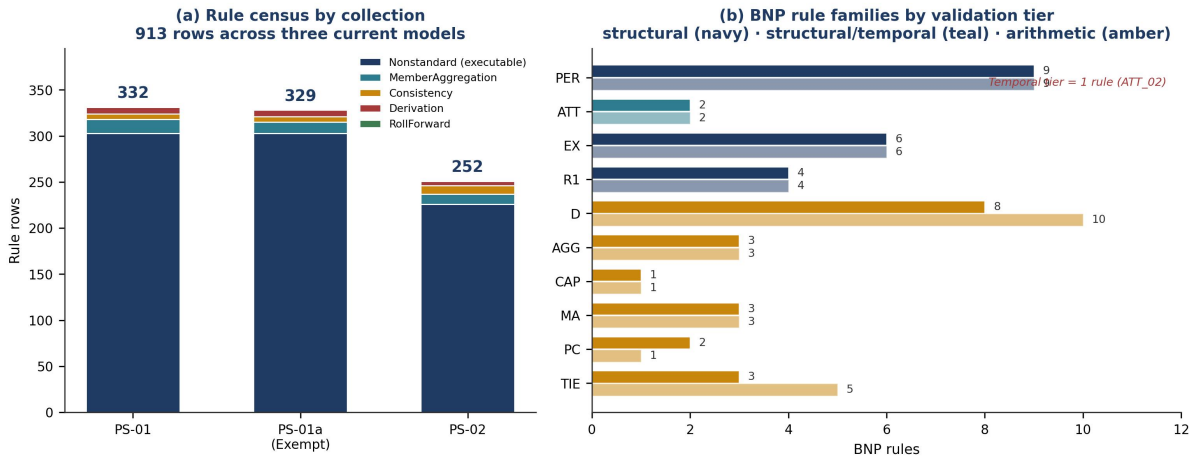

Figure 3: Rule architecture of the current corpus: (a) census by collection; (b) BNP rule families by validation tier

The BNP layer and the three tiers. The 41, 41, and 44 BNP-coded rules decompose

into nine families whose mapping onto the three validation tiers asserted in the regulatory submissions is given in Table 2. The structural tier is implemented in full and considerably beyond its filed description: the affiliate-perimeter machinery of nine rules and the attestation pair were engineered in response to the completeness result of Section 4.3 and appear in no submission. The arithmetic tier is implemented in full, but by value assertions rather than by the calculation linkbase the submissions describe. This is necessary rather than incidental, since calculation arcs cannot express a tolerance, a conditional, or an aggregation across a typed dimension, and no calculation association in any model touches a BNP concept. The temporal tier remains a single rule, `PS01_BNP_ATT_02`, which constrains the attestation date to the window between period end and filing date. No rule compares any BNP metric against a prior period, and the one-percent and five-percent consensus-influence triggers described in the submissions exist as prose, with no parameter and no assertion. The self-assessment indicator for combined consensus influence is collected and bound by nothing.

Table 2: BNP rule families mapped to the filed validation tiers (current models)

Family	**Obligation enforced**	**PS-01 / PS-01a**	**PS-02**	**Tier**
PER	Affiliate perimeter: identification, attested count, zero-case closure, no data outside the perimeter	9	9	Structural
ATT	Attestation completeness and date window	2	2	Structural / temporal
EX	Conditional presence: a reported X compels a Y	6	6	Structural
R1	Indicator-to-quantity coherence	4	4	Structural
D	Share derivation from numerator and network denominator	8	10	Arithmetic
AGG	Combined control equals issuer plus sum of affiliates	3	3	Arithmetic
CAP	Combined slots cannot exceed network capacity	1	1	Arithmetic
MA	Dimensionless total equals sum over the typed axis	3	3	Arithmetic
PC	Percentage facts bounded on the fraction scale	2	1	Arithmetic
TIE	Cross-schedule ties into the income statement	3	5	Arithmetic
Total		**41**	**44**	

Standard versus reduced-burden form. The exemption is not a thinned copy, as Figure 4 shows. The two weekly models share a dictionary to within two abstract headers (681 against 679 terms, with identical concept, member, hypercube, and axis populations) and, in the finding with the sharpest supervisory implication, they share the entire BNP rule set: all 41 rule codes are identical, so the reduced-burden regime relaxes conventional reserve reporting while relaxing no blockchain-participation validation whatever. What differs is the reporting geometry. PS-01a replaces daily line-item detail with weekly statistical summaries, which the model implements not by pruning but by restructuring: the reduced-burden model carries 110 more association rows than the standard model (929 against 819, with 684 presentation arcs against 603), because minimum, maximum, and average statistic trees replace per-day detail and the standalone concentration schedule is folded into the issuer-participation schedule. Its reference instance is correspondingly sparse, at 153 facts against 1,282 and with no members on the affiliate axis, a property whose validation consequence Section 4.4 quantifies.

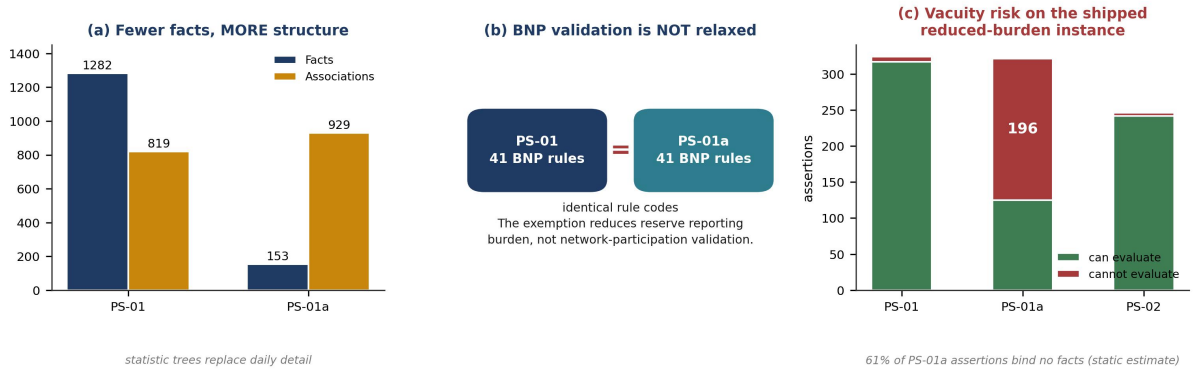

Figure 4: Standard PS-01 against reduced-burden PS-01a: fewer facts but more structure, an identical BNP rule set, and the resulting vacuity exposure

4.2 RQ2: Source-to-Package Fidelity

Figure 5 summarizes the reconciliation of each generated package against its source model. The structural answer is unambiguous: content survives exactly. Instance facts reconcile one-for-one at 1,282, 153, and 771; presentation arcs at 603, 684, and 559; calculation arcs at 9, 9, and 95; and references at 633, 633, and 529. All 41, 41, and 30 hypercubes and all 18, 18, and 15 typed dimensions are declared, with the definition linkbase carrying the hypercube wiring together with the dimension-default and domain-member arcs the specification requires. The apparent surplus of schema elements (699, 697, and 626 declarations against 681, 679, and 611 terms) reconciles exactly to the typed-dimension domain declarations a generator must add. Fact-dimension rows are not lost but re-encoded: the models' 1,106, 97, and 611 per-fact dimension rows normalize into 210, 50, and 134 shared contexts, which is the canonical XBRL representation. Static integrity of the formula linkbase is likewise clean at the rungs static analysis can reach, with no duplicate identifiers, no references to undeclared concepts, no unresolvable variable references, and no malformed delimiters; the only empty test attributes belong to the two existence assertions, for which an empty test is the correct form.

The validation layer, by contrast, is transformed in generation in three ways that matter. First, the generator strips the executable-twin suffix and also compiles the native declarative rows, merging both under a single identifier, which is why the packages carry 324, 321, and 246 assertions against 303, 303, and 226 executable source rows, and which creates the defect documented in Section 4.4. Second, the parameter register explodes correctly into 61, 61, and 59 parameter resources consumed by 699, 693, and 632 fact variables through 1,496, 1,487, and 1,223 filter arcs, so the assertions are richly wired rather than hollow. Third, the formula linkbase is referenced only from the instance and not from the schema, so a consumer who loads the taxonomy alone discovers no rules at all, and the enforcement layer travels with the filing rather than with the standard.

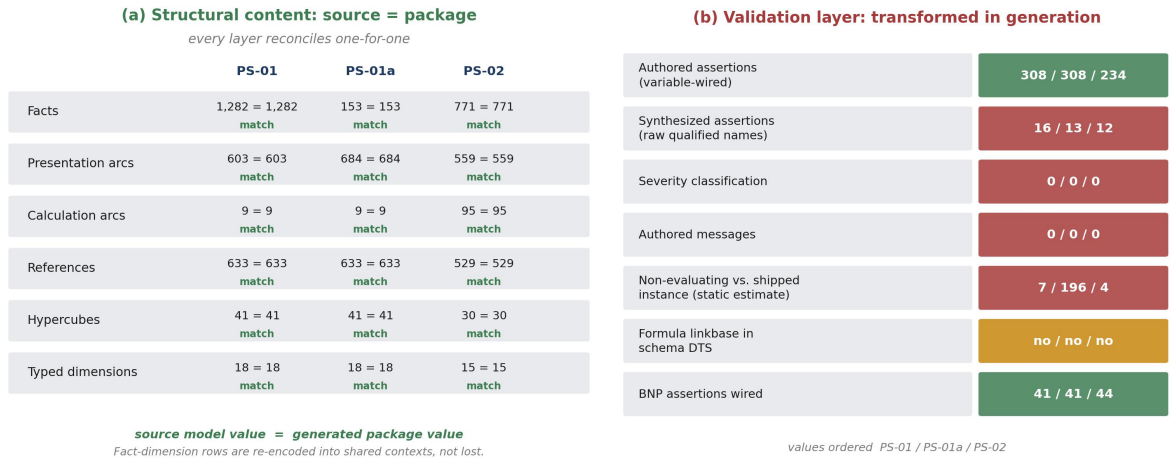

Figure 5: Source-to-package fidelity: (a) structural content reconciles exactly; (b) the validation layer is transformed in generation

4.3 RQ3: Processor-Oriented Evaluation of the Validation Layer

Table 3 classifies every validation-layer deficiency surfaced by the study into the three categories this paper keeps strictly apart.

Current deficiency 1: synthesized assertions. A mechanical signature separates two populations in every delivered linkbase: authored assertions, whose tests are written over declared variables, and synthesized assertions, whose tests are written over raw qualified names. The current packages carry 16, 13, and 12 synthesized assertions out of 324, 321, and 246, and one per package declares no variables at all. The mechanism is identifiable at source: these identifiers exist in the models twice, once as a native declarative row whose expression uses bare names and once as a hand-authored executable twin with correct variables and filters, and the generator's merge keeps the native form. The affected population is concentrated exactly where it matters, in the member-aggregation and cross-total families that carry the roll-up integrity of the filing. Under standard Formula semantics a bare qualified name in a test does not reference the declared variables, and in the project's earlier conformant-processor battery every assertion that failed a compliant filing was of this class while every authored twin passed. All 41, 41, and 44 BNP assertions are authored and variable-wired, and none carries the signature. As a static prediction for the scheduled current-generation run, we state in advance that every unsatisfied result will carry the synthesized signature.

Current deficiency 2: no severity and no messages. Zero severity resources and zero message resources exist in any package. Every rule is therefore an implicit error-level boolean whose only diagnostic is its identifier. A pipeline that flags everything with equal weight has not reduced examiner burden but relocated it, and the models' own commentary records graded intent, describing a rule that should be a warning under one regulatory option and an error under another, with nowhere to express it. Adopting the assertion-severity mechanism and authored messages is the least costly item on the roadmap and disproportionately valuable.[7]

Current deficiency 3: sparse-instance vacuity on the reduced-burden form. Static analysis flags assertions whose concept filters target concepts carrying no facts in the

[7] XBRL International, *Assertion Severity 2.0*, Recommendation. https://specifications.xbrl.org

shipped instance: 7 of 324 on PS-01 and 4 of 246 on PS-02, but 196 of 321 on PS-01a, whose 153-fact instance populates no affiliate-axis members at all. An assertion that binds nothing produces no evaluations, and a pass arising from binding nothing is indistinguishable, in a summary count, from a pass arising from correctness. A clean validation report on the shipped reduced-burden package is therefore largely vacuous, and any production gate over such filings must report evaluation counts and never satisfaction counts alone. This is the operational face of the presence–execution ladder.

Current deficiency 4: discoverability and unbound conduct. The instance-only discovery of the formula linkbase means that taxonomy-level conformance tooling never sees the rules. Separately, the corpus tests quantities and not conduct. The four indicators that would characterize transaction-ordering risk, namely whether ordering decisions are logged, whether an affiliate can influence ordering, whether the combined entities could influence consensus, and whether the filer monitors for front-running, are collected by the taxonomy and bound by no assertion. A filing may therefore affirm affiliate ordering influence, report network revenue, and satisfy every rule in the corpus, because no rule relates them. The framework's eighth domain, covering third-party service providers and constituting its most explicitly anti-evasion tranche, has no typed axis and no rules in any current model.

Historical defects, now fixed. Three findings from the earlier battery are confirmed remediated in the current artifacts and must not be read as present defects. First, the quarterly abort: in four prior observations across two model families, including a fact-less package which localized the defect to the rule layer, one generated assertion carried an escaped copy of an extended link in its test attribute, and the resulting parse failure aborted formula processing for the entire linkbase, so every quarterly package executed zero rules while presenting a clean structural validation. In the current package that assertion carries a clean eighty-character expression, and no assertion in any current linkbase bears the escaped-markup signature. A pipeline whose failure mode is silence is worse than no pipeline, and this discovery with its measurable fix is the clearest demonstration in this study of why regulatory validation must be executed on a conformant processor rather than trusted from self-reports. Second, the dimensional conflict: a concept serving as primary item of a closed typed-axis hypercube cannot validly report a dimensionless total, because typed dimensions have no defaults, and the remediation of a companion totals hypercube per schedule is present in all current models, with the earlier battery measuring the conflicted-concept population at forty-five before and zero after. Residual dimensional validity of the current quarterly package awaits the scheduled run. Third, early shipped linkbases predated the models and contained no BNP-coded assertions, whereas the current linkbases contain all of them.

4.4 RQ4 (Logic): Omission Invariance

The companion paper proposed that an undisclosed affiliate would create a detectable quantitative gap. Evaluation of the current executable rule corpus shows that this claim does not hold under consistent omission, for the reason formalized below. Figure 6 presents the construction.

Proposition 1 (Omission invariance)**.** *Let an assertion be evaluated over operands drawn entirely from a single filing. Then its truth value is invariant to the consistent omission of any party from that filing, that is, to omission from every operand in which the party would have appeared. Consequently no such assertion can distinguish a complete filing from an incomplete one.*

The proof is the observation that every such assertion is a predicate over reported facts, and that a consistent omission is a transformation of the fact set which preserves every

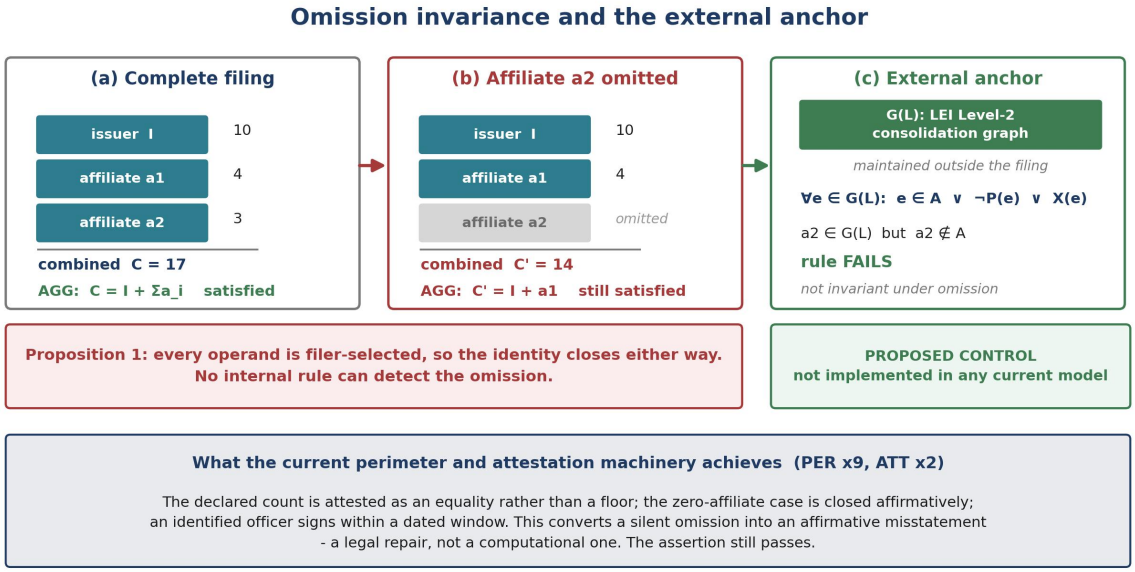

Figure 6: Omission invariance and the external anchor: internal identities close under consistent omission, while an operand outside the filer's control does not

internal relation among the surviving facts. Its bite comes from an exhaustive property of the current corpus, verified by inspection of all 126 BNP rule expressions and confirmed by scanning every rule row in all three models: no rule takes any operand from outside the filing. The aggregation family computes combined control from the issuer's figure plus the sum over precisely the affiliates the filer enumerated, and the combined figure is itself a filer-reported fact, so removing an affiliate from the rows and from the total together leaves the identity closed. The perimeter count rule tests equality between a declared count and the count of declared rows, and both operands move together under omission. The capacity rule bounds combined validator slots by a network total that the filer reports. The existence assertions are floors satisfied by a single disclosure. Even the Legal Entity Identifier is validated at syntax only, through pattern and checksum, so a well-formed identifier absent from the Global LEI System satisfies every identifier check in the corpus. The result is mathematical rather than empirical: it is a property of what the rules can express, established without executing anything, Because this result follows from the operand structure of the rules, runtime execution is not required to establish the proposition.

What the current models did about it deserves precise credit. The nine perimeter rules and two attestation rules, absent from the filed submissions and engineered in response to this result, identify every engaged affiliate by legal name, Legal Entity Identifier, control direction, and control basis; attest the count as an equality rather than as a floor; close the zero-affiliate case affirmatively; and seal the side channel through which an affiliate row carrying metrics but no participation indicator would be summed silently. This construction does not detect an omitted affiliate, and Proposition 1 holds that nothing operating from inside the filing can. What it does is convert a silent gap into an affirmative misstatement of a stated fact, namely the completeness of an enumerated perimeter, made by a named officer on a dated instrument. In the terms of Section 2.3 it removes the Dye condition, since the supervisor no longer faces uncertainty about whether the filer is informed. That is a real gain, and it is a legal gain rather than a computational one. The rule still passes.

Table 3: Validation-layer findings by status

Finding	Status	Evidence
16 / 13 / 12 synthesized assertions; authored twins discarded in the generator merge	Current	Static signature on delivered linkbases; source-level twin comparison
Zero severity; zero authored messages	Current	Element census of all three linkbases
196 of 321 assertions potentially exposed to zero binding in the shipped PS-01a reference instance	Current (static estimate)	Concept filters against instance census; runtime confirmation scheduled
Formula linkbase outside the schema discoverable taxonomy set	Current	Linkbase reference wiring
Four conduct indicators and the third-party domain unbound	Current	Rule-reference scan; axis census
Temporal tier of one date-window rule; consensus-influence triggers unencoded	Current	Rule census
Malformed quarterly assertion aborting all formula processing	Fixed	Clean test in current package; no escaped-markup signatures
Shared-primary-item dimensional conflict (45 concepts)	Fixed	Totals-hypercube remediation present; historical 45-to-0 measurement
Early linkbases carrying no BNP assertions	Fixed	41 / 41 / 44 BNP assertions present and wired
Level-2 anchored perimeter rule; vLEI-bound attestation; accounting crosswalk; consensus-influence triggers	Proposed	No implementing rule exists in any model

4.5 RQ4 (Design): Restoring Detectability

Proposition 1 dictates the shape of any remedy, namely an operand the filer does not control. We advance two, strictly as design propositions.

The Level-2 anchor. Let L be the issuer's Legal Entity Identifier and let $G(L)$ denote the set of entities whose accounting-consolidating parent chain includes L, as published in the Global LEI System's Level 2 relationship data. With A the perimeter declared in the filing, the anchored rule is

$$\forall\, e \in G(L): \quad e \in A \;\vee\; \neg P(e) \;\vee\; X(e) \tag{1}$$

where $P(e)$ asserts that e engages in blockchain network participation and $X(e)$ is a published Level 2 reporting exception. Both operands exist today: A is in the filing and $G(L)$ is a public reference file designated for exactly the receiving agencies. Unlike every rule in the corpus, the test is not invariant under omission, because removing an affiliate from A does not remove it from $G(L)$. Detection fails only inside the enumerated and disclosed exception set, which converts an unknown unknown into an enumerated known. Three boundaries are acknowledged: Level 2 parentage is itself reported, albeit verified by a party other than the filer; enumerated exception categories exist; and an arm's-length third-party validator is by construction outside any consolidation graph, which is why the taxonomy's unimplemented eighth domain, and the verifiable LEI role credential that would bind both the officer's

attestation and the counterparty's identity, complete the construction rather than duplicate it.

The accounting-perimeter crosswalk. A second external operand sits upstream of the filing. The consolidation perimeter of the issuer's financial statements is fixed by ASC 810 or IFRS 10 as applied to the group and tested by an auditor, rather than selected row by row at filing time. An assertion relating the participation schedules to consolidated statements is therefore not omission-invariant, because a party omitted from the schedule persists in the consolidated totals and the two operands cease to agree. The current models already assert this correspondence in prose, referring to the interagency Call Report 91, 91, and 101 times across the three models and naming specific US GAAP elements in the quarterly model, while zero of the 913 rules reference any US GAAP or IFRS element in an executable position. The correspondence is an instruction to a human preparer and is invisible to every processor that will read the filing. We specify four candidate rule families in the supplementary materials and note the standard-setting gap that bounds them: no element in either accounting regime separately identifies network-derived or extractable-value revenue, so the crosswalk must reconcile aggregates in which extractable value is one unlabeled term (Daian et al. 2020; Qin et al. 2022; Auer et al. 2022).

5 Discussion

5.1 The Validation Layer Is an Assurance Object

Continuous auditing has treated the monitored process as the assurance object and the monitoring apparatus as the instrument (Vasarhelyi and Halper 1991; Kogan et al. 2014). The first general lesson of this study is that in machine-cadence supervision the instrument must itself be assured, and with the same discipline: negative controls that prove the gate can report failure, mutation tests that prove each rule fails the defect it was written for, and, decisively, evaluation-count reporting. The quarterly abort and the reduced-burden vacuity share one failure mode, which is silence, and a summary that reports satisfied assertions without reporting evaluations cannot distinguish compliance from a gate that never ran. The presence–execution ladder operationalizes this, since existence, parsing, execution, intended binding, and correct outcome are separate rungs each requiring separate evidence. We propose the ladder as a reporting convention for automated-validation research and for supervisory pipelines alike.

5.2 Fidelity Is a Compilation Problem

The source-to-package reconciliation reframes a common procurement assumption. Regulators specify taxonomies and platforms compile them. Our evidence shows that structural compilation can be exact while validation compilation degrades, with twins merged into their non-executable parents, graded intent lacking a severity channel in which to land, and an enforcement layer discoverable only through the filing it is meant to police. None of this is visible in the source model, and none of it is exotic: it is the ordinary drift of a build pipeline whose outputs nobody independently executes. The remedy is equally ordinary and we state it as policy. The delivered package, and not the internal representation, is the unit of assurance; every quantitative claim in a filing or exhibit should be computed from the shipped artifact by script at the moment of filing; and a load-time gate should fail any package that produces zero evaluations. The XBRL data-quality literature spent a decade documenting errors in what filers said (Debreceny et al. 2010; Du et al. 2013). As prudential regimes go machine-readable, the question for the next decade is errors in what validators do.

5.3 The Boundary Problem Generalizes

Proposition 1 is not about blockchains. Any reporting regime that asks an entity to describe a boundary, whether the affiliate set, the consolidated group, related parties, beneficial owners, or a supply-chain or emissions scope, and then checks that description against itself, has the same structure: both sides of every check are computed from the population the entity chose to report, so the checks are invariant under consistent omission. Machine-readability does not change this and only makes the self-referential checking fast. What changes it is an operand outside the reporting entity's control, and the available anchors differ by regime, whether the Level 2 consolidation graph used here, registry data, counterparty filings, or supervisory cross-filing comparison elsewhere. The disclosure-theoretic reading is exact: internal validation verifies the message, attestation establishes that the sender is informed (Dye 1985), and only external anchoring restores the verifiability condition under which unravelling disciplines the boundary itself (Grossman 1981; Milgrom 1981). For the assurance literature, this locates completeness, the assertion auditors have always found hardest, as precisely the property that cannot be automated from within the reporting artifact, however sophisticated the rule language.

5.4 Implications for Stablecoin Supervision

Three operational conclusions follow for the GENIUS Act build-out now underway. First, the BNP layer is real and agency-invariant, being identical across the standard and reduced-burden weekly forms, which supports treating network-participation validation as a common supervisory module rather than as a per-form artifact. Second, the reduced-burden regime reduces data and not validation; but a rule set evaluated against data that is not collected is a set of assertions with nothing to bind, and the vacuity result quantifies how far a clean small-issuer filing can be from a checked one. Third, the corpus tests quantities and not conduct: every monetary participation fact is bound by some assertion and every indicator of transaction-ordering influence is bound by none, which is the precise inversion of the extraction-centred supervisory concern the companion paper documents (El Imami et al. 2026; Auer et al. 2022; Materwala et al. 2025). Encoding even existence-level ties from conduct indicators to revenue facts would close the least costly half of that gap, while the other half awaits an accounting element that names extractable value.

These findings suggest that machine-readable supervision should distinguish three assurance objects: the source model, the delivered XBRL package, and the validation outcome produced from that package. Structural fidelity at the first two levels does not establish that the rules executed or bound the intended facts. For supervisory use, validation reports should therefore disclose not only satisfied and unsatisfied assertions, but also assertions discovered, evaluation counts, zero-binding conditions, severity, and diagnostic messages. A delivered package producing zero evaluations should fail the supervisory intake process rather than appear compliant.

The findings also distinguish internal consistency controls from completeness controls. Arithmetic, derivation, aggregation, and dimensional rules can establish whether reported facts agree with one another, but they cannot establish that the filer reported the entire population when that population is itself filer-defined. Perimeter attestation improves accountability by turning silent omission into an affirmative representation, but computational detection requires an independently maintained reference point. For stablecoin supervision, this implies a two-layer architecture: internal XBRL validation for consistency and external anchoring for reporting-boundary completeness.

6 Limitations and Future Research

Six limitations bound the claims. First, the reference instances were constructed by the project that wrote the rules and are compliant by construction, so universal passes establish that rules bind and do not fire spuriously rather than how they behave against filings prepared by parties with an interest in the answer; no real issuer filings yet exist under these forms. Second, the runtime boundary is explicit: all conformant-processor results cited here belong to earlier generations and are labelled as such, the current-generation packages have been statically audited in full, and their execution with per-assertion evaluation counts is the immediate next step, with the static predictions of Section 4.4 stated in advance as falsifiable. Third, the zero-binding figure for the reduced-burden form is a static estimate derived from concept filters against the shipped instance, and runtime will refine it, though not its order of magnitude. Fourth, the external anchors are propositions with proofs of non-invariance rather than implementations with results; the practical force of the Level 2 anchor is an empirical coverage question, namely what fraction of participation-relevant affiliates carry reported parentage rather than exceptions, which is answerable today from public reference files and which we regard as the highest-value next study alongside a systematic adversarial-filing exercise that would generalize Proposition 1 from an analytical result into a mapped attack surface. Fifth, the corpus audited implements seven of the framework's eight domains, so statements about what the rules enforce apply to the implemented seven-eighths. Sixth, generalization beyond this regime is argued structurally, through the premises of Proposition 1, rather than demonstrated empirically in other regimes; comparative work, most naturally against the validation logic of the European Union's Markets in Crypto-Assets machine-readable disclosures, would test whether regimes that agree on syntax can be made to agree on constraints.

7 Conclusion

A machine-readable disclosure taxonomy is necessary and not sufficient. Between the taxonomy a regulator adopts and the enforcement it expects the reporting architecture to provide lie two distinct gaps that this paper measures. The first is an engineering gap: source models survive compilation structurally intact while their validation layer degrades, with assertions synthesized past their authored twins, no severity and no messages, an enforcement layer that travels only with the filing, and, until its recent remediation, a single malformed rule that silenced every quarterly filing while presenting a clean report. The second is a logical gap that no engineering closes. Wherever a regime asks an entity to describe its own boundary and then checks that description against itself, the boundary is whatever the entity says it is, because every check is invariant under consistent omission. The attestation machinery in the current models converts that silence into an actionable misstatement, which is a genuine legal advance; detection awaits an operand the filer does not control, and the consolidation graph of the Global LEI System and the audited accounting perimeter are two such operands already standing. Machine-readable disclosure does not establish completeness. It makes the checking fast, and, done rightly, it makes visible at supervisory cadence where internal checking ends and external anchoring must begin.

Acknowledgments: We extend our thanks to the entire team at the CARLab of Rutgers Business School as well as the National Center for Scientific and Technical Research in Rabat. Special thanks to our Auditchain colleagues for their constant support.

Conflict of interest: Authors affiliated with Auditchain Labs AG participated in the development of the artifacts studied. The evaluation protocol was designed so that every figure

reported in this paper is independently recomputable from the shipped artifacts.

References

Alles, Michael et al. (2006). "Continuous monitoring of business process controls: A pilot implementation of a continuous auditing system at Siemens". In: *International Journal of Accounting Information Systems* 7.2, pp. 137–161. DOI: 10.1016/j.accinf.2005.10.004.

Appelbaum, Deniz, Alexander Kogan, and Miklos A. Vasarhelyi (2017). "Big data and analytics in the modern audit engagement: Research needs". In: *Auditing: A Journal of Practice & Theory* 36.4, pp. 1–27. DOI: 10.2308/ajpt-51684.

Auer, Raphael, Jon Frost, and Jose María Vidal Pastor (2022). "Miners as intermediaries: Extractable value and market manipulation in crypto and DeFi". In: *BIS Bulletin* 58.

Bartley, Jon, Al Y. S. Chen, and Eileen Z. Taylor (2011). "A comparison of XBRL filings to corporate 10-Ks: Evidence from the voluntary filing program". In: *Accounting Horizons* 25.2, pp. 227–245. DOI: 10.2308/acch-10028.

Belenkov, Nikita et al. (2025). "SoK: A review of cross-chain bridge hacks in 2023". In: *arXiv preprint arXiv:2501.03423*. DOI: 10.48550/arXiv.2501.03423.

Beyer, Anne et al. (2010). "The financial reporting environment: Review of the recent literature". In: *Journal of Accounting and Economics* 50.2-3, pp. 296–343. DOI: 10.1016/j.jacceco.2010.10.003.

Blankespoor, Elizabeth, Brian P. Miller, and Hal D. White (2014). "Initial evidence on the market impact of the XBRL mandate". In: *Review of Accounting Studies* 19.4, pp. 1468–1503. DOI: 10.1007/s11142-013-9273-4.

Boritz, J. Efrim and Won Gyun No (2009). "Assurance on XBRL-related documents: The case of United Technologies Corporation". In: *Journal of Information Systems* 23.2, pp. 49–78. DOI: 10.2308/jis.2009.23.2.49.

Daian, Philip et al. (2020). "Flash Boys 2.0: Frontrunning in decentralized exchanges, miner extractable value, and consensus instability". In: *2020 IEEE Symposium on Security and Privacy*. IEEE, pp. 910–927. DOI: 10.1109/SP40000.2020.00040.

Debreceny, Roger et al. (2010). "Does it add up? Early evidence on the data quality of XBRL filings to the SEC". In: *Journal of Accounting and Public Policy* 29.3, pp. 296–306. DOI: 10.1016/j.jaccpubpol.2010.04.001.

Debreceny, Roger S. et al. (2011). "Flex or break? Extensions in XBRL disclosures to the SEC". In: *Accounting Horizons* 25.4, pp. 631–657. DOI: 10.2308/acch-50068.

Diop, Papa Ousseynou, Julien Chevallier, and Bilel Sanhaji (2024). "Collapse of Silicon Valley Bank and USDC depegging: A machine learning experiment". In: *FinTech* 3.4, pp. 569–590. DOI: 10.3390/fintech3040030.

Du, Hui, Miklos A. Vasarhelyi, and Xiaochuan Zheng (2013). "XBRL mandate: Thousands of filing errors and so what?" In: *Journal of Information Systems* 27.1, pp. 61–78. DOI: 10.2308/isys-50399.

Dye, Ronald A. (1985). "Disclosure of nonproprietary information". In: *Journal of Accounting Research* 23.1, pp. 123–145. DOI: 10.2307/2490910.

Eichengreen, Barry, My T. Nguyen, and Ganesh Viswanath-Natraj (2025). "Stablecoin devaluation risk". In: *The European Journal of Finance* 31.11, pp. 1469–1496. DOI: 10.1080/1351847X.2025.2505757.

El Imami, Imane et al. (2026). "Beyond the fiat vault: Evaluating machine-readable blockchain disclosures for systemic safety, operational security, and MEV conflict mitigation". In: *SSRN Electronic Journal*. DOI: 10.2139/ssrn.6837058. URL: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=6837058.

Grossman, Sanford J. (1981). "The informational role of warranties and private disclosure about product quality". In: *The Journal of Law and Economics* 24.3, pp. 461–483. DOI: 10.1086/466995.

Hoitash, Rani and Udi Hoitash (2018). "Measuring accounting reporting complexity with XBRL". In: *The Accounting Review* 93.1, pp. 259–287. DOI: 10.2308/accr-51762.

Kim, Joung W., Jee-Hae Lim, and Won Gyun No (2012). "The effect of first wave mandatory XBRL reporting across the financial information environment". In: *Journal of Information Systems* 26.1, pp. 127–153. DOI: 10.2308/isys-10260.

Kogan, Alexander et al. (2014). "Design and evaluation of a continuous data level auditing system". In: *Auditing: A Journal of Practice & Theory* 33.4, pp. 221–245. DOI: 10.2308/ajpt-50844.

Materwala, Huned, Leila Ismail, and Hossam S. Hassanein (2025). "Maximal extractable value in decentralized finance: Taxonomy, detection, and mitigation". In: *IEEE Transactions on Services Computing* 18.6, pp. 4386–4407. DOI: 10.1109/TSC.2025.3620604.

Milgrom, Paul R. (1981). "Good news and bad news: Representation theorems and applications". In: *The Bell Journal of Economics* 12.2, pp. 380–391. DOI: 10.2307/3003562.

Notland, Jakob Svennevik et al. (2026). "SoK: Cross-chain bridging architectural design flaws and mitigations". In: *Blockchain: Research and Applications* 7.1, p. 100315. DOI: 10.1016/j.bcra.2025.100315.

Qin, Kaihua, Liyi Zhou, and Arthur Gervais (2022). "Quantifying blockchain extractable value: How dark is the forest?" In: *2022 IEEE Symposium on Security and Privacy*. IEEE, pp. 198–214. DOI: 10.1109/SP46214.2022.9833734.

Vasarhelyi, Miklos A. and Fern B. Halper (1991). "The continuous audit of online systems". In: *Auditing: A Journal of Practice & Theory* 10.1, pp. 110–125.

Verrecchia, Robert E. (1983). "Discretionary disclosure". In: *Journal of Accounting and Economics* 5, pp. 179–194. DOI: 10.1016/0165-4101(83)90011-3.