ZETTL IV: Payment Facts, Typed Decisions and Review

A proposed applications layer: a compliance and analytics dashboard and a calibrated payment decision model

Jeet Raut, ZETTL

v0.5r4 — proposal, 15 October 2026

v0.5r4 — proposal, 15 October 2026. This paper is new in v0.5r4. It is a research and product proposal, not a description of working software. Nothing in it is implemented, measured or validated with customers. It gives no training-data access, customer commitments, accuracy, savings, costs, security guarantees or jurisdictional coverage, because none exists.

This paper is not repository authority, and it does not change any technical contract of Papers I–III. Status markers used throughout:

- [Accepted v0.4] an accepted protocol design contract (not implemented);
- [Direction, 15 Oct 2026 (v0.5r4)] the program direction of 15 October 2026;
- [Decided v0.5 direction] a decided v0.5 protocol direction; [Proposed, pending Z01] a protocol proposal pending Z01;
- [Product, off-protocol] a proposed product, outside the protocol;
- [Research hypothesis] a research hypothesis to be tested;
- [External dependency] a third-party dependency whose terms can change;
- [Unsupported, needs evidence] a claim that would need evidence Zettl does not have.

Abstract

Papers I–III of this series specify a settlement network: unanimous-certificate settlement, a privacy lane and interoperability for agent payments. This paper proposes an applications layer above them and a plan for learning whether it is worth building. It covers two connected product efforts.

The first is a *cross-jurisdictional compliance, regulatory-flags and analytics dashboard*. It imports payment records and their supporting evidence and normalizes them into payment facts. It shows each payment's economic purpose, relationships and state, surfaces missing facts and possible review flags, and supports human review with an audit history.

The second is a *payment decision model*, which we call *Zlef*. Inspired by TypeSafe's Jev and Cloudflare's Clef, it has typed, bounded outputs with probabilities; we add an explicit way to say "unknown", which neither vendor states. Zlef would classify economic purpose, relationships and evidence completeness; transaction mechanics are read from records.

Three separations run through the design:

- facts the protocol can authenticate, facts a source merely attests, absent facts, model predictions, rule matches and reviewed conclusions;
- calibrated scores and model-generated explanations;
- economic purpose, transaction mechanics and jurisdiction-specific legal or tax treatment, which stays with versioned rules and human review.

We define the model's inputs, outputs, trust boundary, calibration objectives, abstention and evaluation methodology, and a seven-step evaluation sequence with preregistered decisions. Nothing here is implemented or measured. Whether a buyer exists is the first question, and protocol progress does not answer it.

Series. This is Paper IV of the ZETTL series:

I: *Unanimous-Certificate Settlement with Ordered Recovery*;

II: *Shielded Payments with Ordered Settlement*;

III: *Interoperability and Agent Payments*;

IV: *Payment Facts, Typed Decisions and Review* (a proposed applications layer);

V: *Cross-Jurisdictional Payment Compliance* (its rules layer, v0.5r5);

VI: *Zlef: Typed, Calibrated Decisions over Crypto-Asset Payments* (its model layer, v0.5r5).

Papers I–III state protocol designs. This paper states product and research proposals that use them. The series will be combined into a single paper ahead of mainnet.

Contents

1 Introduction **3**

2 Background **4**

3 Problems, Buyers and the PMF Track **5**

4 Architecture of the Applications Layer **6**

5 The Payment Decision Model **8**

6 Rules, Review and Audit **10**

7 Evaluation Methodology **11**

8 What the Protocol Contributes **12**

9 Roadmap and Open Decisions **13**

10 Conclusion **14**

A Claim and Status Matrix **15**

1 Introduction

ZETTL began as a settlement network. Papers I–III state its design: certificates signed by every validator, ordered recovery, a privacy lane in four levels, and receipts and light clients for agent payments. None of that is implemented, and none of it tells us whether anyone will pay for something built on it.

Direction ([Direction, 15 Oct 2026 (v0.5r4)]). On 15 October 2026 Jeet set the program's priorities as three separate tracks:

1. **PMF experiments**, the immediate priority. They identify a specific buyer, a costly workflow problem, the existing workaround, the adoption friction and the willingness to pay.
2. **Product viability**. It establishes whether a narrow product delivers value with acceptable integration effort, trust assumptions, reliability and cost.
3. **Continuing protocol development**, under the existing acceptance gates of Papers I–III.

Two rules keep the tracks honest. Protocol progress does not establish product-market fit, and customer interest does not establish protocol correctness. Every milestone in this paper names the track it belongs to.

What this paper proposes. Two product efforts developed in parallel, and the evaluation plan that decides whether to continue, narrow, change or stop them:

- a compliance, regulatory-flags and analytics dashboard (Section 4) [Product, off-protocol];
- a typed payment decision model, Zlef, designed to be calibrated (Section 5) [Product, off-protocol]. Whether it outperforms simpler baselines on a chosen workflow is a [Research hypothesis].

How this relates to v0.5r3. The v0.5r3 papers named pay-for-result settlement for agent and API sellers the company's first product. This revision changes that priority. Agent and API sellers remain one candidate audience, and the pay-for-result design in Paper III is unchanged. The immediate PMF work now centres on the dashboard and the model. Private payouts remain a gated candidate (Paper II). Accounting-suite and medical-billing concepts are separate discovery projects. Their taxonomies, sensitive-data requirements and rules are not merged with this one. None of these audiences is a validated customer segment.

What this paper is not. It is not a claim of comprehensive compliance, legal certainty or automated tax correctness. It does not claim that a model label determines legal or tax treatment, and it does not claim that Zettl has implemented RLCD. It is not evidence that the dashboard and the model together form a moat. That is a [Research hypothesis].

What it authorizes. This paper is document preparation. It does not authorize outreach, deployment, real payments, training expenditure, or the resumption of technical work that was specifically stopped. Each keeps its own gate.

Organization. Section 2 reviews decision models and calibration. Section 3 states candidate problems and buyers. Section 4 gives the architecture and its trust boundary, Section 5 the decision model and Section 6 the rules and review layer. Section 7 is the evaluation methodology and Section 8 the connection to the protocol. Section 9 gives the roadmap and the open decisions, and Appendix A the claim and status matrix.

2 Background

2.1 Decision models

[External dependency] A recent class of models answers typed questions about unstructured state instead of generating text. TypeSafe AI's Jev, announced on 15 September 2026, is described as producing "type-safe structured values" whose "possible outputs and structure are defined in advance". Its launch post states that "all answers are accompanied with calibrated probabilities and confidence scores" [9]. It is offered as an early-access API. The launch post mentions neither open weights nor an abstain option.

Cloudflare's Clef decision models were announced on 1 October 2026. Cloudflare writes: "We currently use Qwen as the base model and post-trained it to suit decision model use cases." Post-training "utilizes label-smoothed cross-entropy for valid schema outputs paired with a Brier loss". Cloudflare "also developed Reinforcement Learning for Calibrated Decisions (RLCD) to serve as a secondary optimization target", and describes its reward as giving partial credit to adjacent ordinal choices. The models are published on Hugging Face under an Apache 2.0 licence and served on Workers AI [2].

Both vendors report benchmarks, and Cloudflare's own table is mixed. These are vendor results on vendor-chosen tasks. They are not evidence about payment-domain performance, and nothing in this paper relies on them.

How we use the term RLCD. RLCD is Cloudflare's name for one of its own training objectives. We use it only that way, and never as a synonym for classification. Because its reward credits adjacent *ordinal* choices, an RLCD-style objective is a candidate for ordinal outputs such as review priority or evidence completeness. Its value for nominal categories such as economic purpose is an open question (Section 7.1).

2.2 Calibration and abstention

A score is *calibrated* when, among the cases it scores at p, a fraction close to p are correct. Modern neural classifiers are often miscalibrated [7]. Standard tools are reliability diagrams, expected calibration error and the Brier score [1]. Abstaining when a prediction is too uncertain is the reject option [3]. Selective classification trades coverage against error rate on the cases kept [6].

Two more concerns shape the evaluation in Section 7. When errors have unequal costs, the decision rule should weigh them [4]. Leakage between training and evaluation data inflates results [8], and drift in data and rules degrades them over time [5].

2.3 Compliance analytics today

We make no market claim here. Which buyers exist, what they use now and what it costs them are the questions of the PMF track (Section 3).

3 Problems, Buyers and the PMF Track

3.1 The problem, stated as a hypothesis

[Research hypothesis] Businesses and the professionals who serve them receive or send payments that cross jurisdictions and rails. For some of them, understanding those payments is a costly, recurring workflow. The work means establishing three things:

- what each payment was for;
- how it relates to invoices, contracts, refunds and other payments;
- what facts are missing, and which payments may need review under which rules.

Today that work is done by hand or with general tools. Whether this is true for a specific buyer, how much it costs them, and whether they would pay to reduce it are unknown.

3.2 Candidate buyers and workflows

The following are candidates to test. None is validated, and none is chosen yet (Section 9.1):

- finance and operations teams at crypto-native organizations that reconcile stablecoin payouts to contributors, grantees and vendors;
- accounting and tax practices whose clients hold or transact in digital assets;
- payment and treasury teams at businesses that accept or pay in stablecoins across borders;
- agent and API sellers that take x402 or MPP payments (the v0.5r3 audience, Paper III).

Accounting-suite and medical-billing concepts are separate discovery projects. They have different taxonomies, regulated data (medical billing especially) and rules, and this paper does not cover them.

3.3 How the PMF track learns before anything moves money

Demand discovery does not wait for the settlement stack. In rough order of cost, the PMF track uses:

1. **structured interviews**, which ask about a recent instance of the workflow, the time and people it took, the current workaround and tools, what goes wrong, who signs off, and what they pay now;
2. **synthetic demonstrations**: a dashboard walkthrough on clearly synthetic records;
3. **historical-record analysis** on the buyer's own records, with permission, a written data agreement and minimum necessary data;
4. **offline verification**, which re-checks a sample of the buyer's past reviews against what the dashboard would have flagged;
5. **integration walkthroughs** that map the buyer's export formats, systems and approval chain.

A money-moving pilot, and any deployment, keeps its own gate.

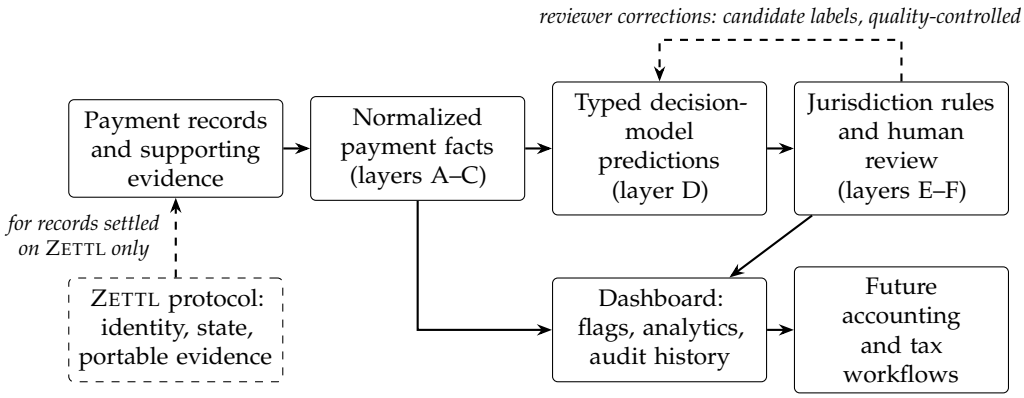


Figure 1: The proposed pipeline. Solid arrows are the initial product, which can run on imported or synthetic records. The dashed protocol box contributes authenticated facts only for records settled on ZETTL. The dashed feedback arrow carries reviewer corrections back as candidate labels, which pass quality control before training or evaluation. Proposed; nothing is built.

Evidence of PMF. Willingness to pay or commit, observed in a payment, a signed pilot or a reallocated budget, and repeated use of a walkthrough on real work. Interest alone is not evidence of PMF, and protocol progress is never evidence of PMF.

Preregistered decisions. Before each experiment we write down what we will measure and the result that would make us continue, narrow, change or stop. We invent no thresholds here. They are agreed before the experiment runs (Section 7).

4 Architecture of the Applications Layer

[Product, off-protocol] Figure 1 shows the proposed pipeline. The dashboard is the workflow interface. The decision model is a reusable interpretation component. The protocol of Papers I–III is the broader infrastructure, and the initial product does not depend on it.

4.1 Fact layers and the trust boundary

Every value the dashboard shows carries the layer it comes from (Table 1). The layers are never merged into one "truth" column. Layer A is empty today: no ZETTL software runs, so no record yet carries protocol authentication.

Non-inferences. The system never infers:

- legal jurisdiction from wallet or network location;
- residency or entity status from a blockchain address;
- legal or tax treatment from an economic-purpose label;
- delivery, subjective quality or tax treatment from a signed claim;
- ZETTL cryptographic provenance for an imported record.

	Layer	Example	Who stands behind it
A	Protocol-authenticated fact	This payment settled on ZETTL with a certificate; amount, asset, accounts; a request-digest binding only if the Z01 proposal is adopted	Cryptography and the accepted contracts (Paper I), for ZETTL records only; none exist yet
B	Source-attested fact	An imported bank, processor or ERP export row, with its source and a hash of the file	The source system; integrity of the import, not truth
C	Absent fact	No invoice linked; counterparty entity type unknown	Nobody (shown as missing)
D	Model prediction	Economic purpose: refund, 0.81; evidence complete: no	The model, at its evaluated calibration
E	Rule match	Flag from rule r (version, source, effective date)	The rule owner
F	Reviewed conclusion	"Refund confirmed; linked to invoice 1042", by reviewer, date	The reviewer, with audit history

Table 1: Fact layers. Example values are illustrative.

4.2 Dashboard capabilities

[Product, off-protocol] The proposed capabilities are:

- importing and normalizing payment records and their supporting evidence: invoices, contracts, counterparty and entity information, asset and network information, and payment evidence;
- showing economic purpose, transaction relationships (refunds, reversals, linked obligations) and payment state;
- surfacing jurisdictional exposure, missing information and potential review flags, each linked to its facts, evidence, rule version, source and effective date where available;
- human review, with corrections, exception handling and an audit history;
- analytics, and structured exports that can later feed accounting and tax software.

The initial scope is one buyer, one workflow and one or two jurisdictions (Section 9.1). Coverage of every jurisdiction is [Unsupported, needs evidence] and is not claimed.

4.3 Data handling

Inputs may include personal and commercial data. The proposal assumes:

- permissioned data only, under a written agreement;
- minimum necessary fields;
- explicit retention and residency;
- separation of customers' data.

Whether data may leave the customer's environment, for example to a third-party model API, is a decision for each customer (Section 5.6). No security guarantee is claimed.

5 The Payment Decision Model

[Product, off-protocol] The decision model, which we call *Zlef*, interprets normalized payment facts into typed, probabilistic predictions. It is inspired by typed decision models such as Jev and Clef (Section 2) in three respects only: outputs are typed and bounded, every output carries a probability, and decisions are not free-text generations. Zettl does not use either model. Zlef is a working name for a proposed model; it is not a Cloudflare product and not built. Whether a specialized model beats simpler baselines on a chosen workflow is a [Research hypothesis] (Section 7).

5.1 Inputs

The model sees normalized facts from layers A–C (Table 1) and references to evidence, never more than the question needs:

- amount, asset, network or rail, timestamps and payment state;
- counterparty identifiers as the source records them, and any entity information the customer supplies;
- memo, reference and invoice text, when the customer permits it;
- links to related records (prior payments, refunds, invoices), each with its layer;
- an explicit list of absent facts (layer C).

Each input carries its layer. A source-attested value is never presented to the model as protocol-authenticated.

5.2 Typed outputs

Every question is defined in advance, with a fixed output type. An illustrative schema, to be replaced by the taxonomy chosen with the first buyer (Section 9.1):

Question	Type	Values
Economic purpose	choice (nominal)	service purchase, refund, internal transfer, fee, payroll, loan, distribution, UNKNOWN
Relationship	choice (nominal)	stand-alone, refund of x, reversal of x, instalment of x, UNKNOWN
Evidence sufficient for purpose?	yes/no	plus UNKNOWN
Evidence completeness	score (ordinal)	0–4
Review priority	choice (ordinal)	low, medium, high, urgent

Economic purpose, transaction mechanics and legal or tax treatment are separate layers. The model predicts purpose and relationships. Transaction mechanics are read from records, not predicted. It never outputs a legal or tax treatment; that comes from versioned rules and review (Section 6).

5.3 Outputs, abstention and explanations

For each question the model returns a probability distribution over the allowed values, and the system may *abstain*. Abstention is decided in two ways, both evaluated (Section 7):

- an explicit UNKNOWN value, which the model can predict;
- a reject option [3, 6]: if the top probability falls below a threshold chosen on validation data for the workflow's error costs [4], the item goes to review.

An abstention is a normal outcome, shown to the reviewer as "model did not decide", not an error.

Explanations are kept apart from scores. The dashboard may show the facts that a prediction depended on, or a short generated rationale, but a rationale is display material. It is never used to compute, adjust or justify a probability, and it is never presented as the reason the model decided.

5.4 Calibration objective

We want probabilities that mean what they say: among items predicted at 0.8, about 80% should be right [7]. The objectives considered are proper scoring rules (log loss, Brier score [1]) and post-hoc calibration such as temperature scaling [7], measured per category, not only in aggregate. Calibration is measured on a held-out set and re-measured over time, because it drifts [5].

5.5 Training and optimization, if justified

Training is not authorized by this paper. If evaluation (Section 7) shows that baselines fall short in a way that training could fix, the options, in increasing cost, are:

1. better rules and prompts for an existing model;
2. calibration of an existing model's outputs;
3. supervised fine-tuning of an open-weight base on permissioned labels;
4. an RLCD-style secondary objective, as Cloudflare describes for Clef [2].

RLCD as described gives partial credit to adjacent ordinal choices. It fits ordinal questions here (evidence completeness, review priority). Whether, and how, it applies to nominal questions such as economic purpose is an open question (Section 7.1). Zettl has not implemented RLCD.

Reviewer corrections are *candidate* labels, not truth. They pass quality control (double review on a sample, disagreement tracking, reviewer-level checks) before they enter training or evaluation data, and corrections made after seeing the model's suggestion are recorded as such, because they may be anchored on it.

5.6 Hosting options and data permission

Option	Data leaves customer?	Status
Rules only	No	Baseline
Jev via API [9]	Yes, to TypeSafe	[External dependency]; early access
Clef via Workers AI [2]	Yes, to Cloudflare	[External dependency]
Clef weights, self-hosted (Apache 2.0)	No, if hosted by customer	[External dependency] weights; hosting effort unknown
General LLM, structured output	Depends on provider	Baseline

Whether payment data may go to a third-party API is the customer's decision under their agreements. Vendor availability, pricing and terms can change.

5.7 Adversarial inputs

Memo and invoice text is attacker-controllable: a counterparty can write "refund" in a memo to change a label, or embed instructions aimed at a language model. The model treats text fields as data. Rule flags never depend on a model label alone for high-cost outcomes, and the robustness of the model to manipulated text is a research question (Section 7.1).

6 Rules, Review and Audit

[Product, off-protocol] Legal and tax treatment are determined by a versioned rules layer and human review. They are not a theorem of this paper and not an output of the model.

6.1 Rule records

Each rule is a versioned record with:

- the jurisdiction or jurisdictions it applies to;
- its condition, over facts (layers A–C), model predictions (layer D) and their probabilities;
- the flag or outcome it produces;
- the source it encodes (statute, regulation, guidance or the customer's policy), with a citation;
- its effective date range, version and owner.

Rule texts are [External dependency]. Who writes and signs off rules (in-house, a licensed partner, or the customer's advisers) is an open decision (Section 9.1).

A *rule match* (layer E) means the rule's condition held on the inputs shown. It is not a legal conclusion. A flag that depends on a model prediction records the prediction and its probability, so a reviewer can see when a flag rests on an uncertain label.

6.2 Review

A reviewer sees, for each flagged or abstained item:

- the facts, each tagged with its layer, and what is missing;
- the model's predictions with probabilities, or that it abstained;
- the rule that fired, its version, source and effective date;
- the evidence links.

The reviewer confirms, corrects or escalates. The result is a reviewed conclusion (layer F) with the reviewer, the time, and the rule and model versions in force. Nothing is overwritten: the audit history keeps every prior state.

6.3 Exports

Exports carry the layer of every value, so downstream accounting or tax software can tell a reviewed conclusion from a prediction. Integration with specific accounting or tax software is future work and is not claimed.

7 Evaluation Methodology

[Product, off-protocol] The evaluation runs in seven steps. Each step's result decides whether the next one is worth doing. No numerical thresholds are given here; they are preregistered with the first buyer before each experiment.

1. **Scope.** Choose one buyer and workflow. Define the taxonomy and the decision schema (Section 5.2) with them.
2. **Data.** Assemble a permissioned, labeled evaluation set. Record who labeled each item, how, and with what disagreement.
3. **Baselines.** Build a thin dashboard. Compare simple rules, an existing decision model (Jev or Clef) and a general model with structured output, on the same items (Section 5.6).
4. **Metrics.** Measure, per category:
   - precision and recall;
   - calibration (reliability diagrams, expected calibration error, Brier score [1, 7]);
   - abstention and coverage, as a risk–coverage curve [6];
   - reviewer correction rate and review time;
   - latency and cost per item;
   - the cost of each error type, since a missed flag and a false flag cost different amounts [4].
5. **Splits and drift.** Split by customer and by time, not at random, to avoid leakage [8]. Keep a held-out set that is never used for tuning. Version data, models and rules, and track drift [5].
6. **Optimization, only if justified.** Fine-tune, or use RLCD-style optimization, only if a measured gap justifies it and spending is separately authorized. Human corrections are candidate labels, not truth.
7. **Decide.** Apply the preregistered decision: continue, narrow, change or stop.

What counts as a result. A result on synthetic data is reported as synthetic. A vendor benchmark is not a result for this domain. A result on one customer is not reported as general.

7.1 Research questions

RQ1. *Calibration.* Are per-category probabilities calibrated on held-out customers and later time periods, and how fast do they drift?

RQ2. *Abstention.* Which abstention mechanism (an explicit UNKNOWN class or a reject threshold) gives a better risk–coverage trade-off for the workflow's error costs?

RQ3. *Ordinal versus nominal.* RLCD as described rewards adjacent ordinal choices. Does an analogous objective help nominal questions, and what would "adjacent" mean there?

RQ4. *Baselines.* Does a specialized model beat rules and general models enough to justify its cost? [Research hypothesis]

RQ5. *Leakage.* How much do random splits overstate performance relative to customer and time splits?

RQ6. *Adversarial text.* How easily do manipulated memos or invoices change labels or flags?

RQ7. *Label quality.* How much do anchored reviewer corrections bias training data?

RQ8. *Privacy.* Which questions can be answered with minimum fields, or with L3 disclosures instead of full records (Paper II)?

RQ9. *Provenance.* Does layer-A provenance for ZETTL-settled records measurably reduce review effort? [Research hypothesis]

8 What the Protocol Contributes

ZETTL's contribution to the applications layer is payment identity, payment state, portable evidence and settlement. Only records that settled on ZETTL would carry it; none exist today, because nothing is implemented.

From Paper I. [Accepted v0.4] A UF certificate shows that every validator signed a given payment; an ordered outcome shows its canonical result. These authenticate the amount, asset and accounts (layer A). A request-digest binding is [Proposed, pending Z01] pending Z01 and would be authenticated only once adopted. They do not authenticate purpose, the counterparty's identity in the world, delivery, or tax treatment.

From Paper II. [Decided v0.5 direction] An L3 disclosure is user-held and selective. A customer can supply one as permissioned evidence for a specific payment. Disclosures reveal what the holder chooses; the dashboard cannot see shielded payments that were not disclosed, and analytics over the private lane are limited to what is disclosed.

From Paper III. [Product, off-protocol] Signed responses, vouchers and receipts are evidence of what a party signed. Only equivocation and duplication are provable. A receipt does not prove delivery, quality or tax treatment.

Imported records. Bank, processor, ERP and other chains' records are layer B. The dashboard never presents them as ZETTL-authenticated.

Independence. The initial product runs on imported or synthetic records and does not wait for the protocol. Whether the dashboard and the protocol reinforce each other, for example by making ZETTL-settled payments cheaper to review, is a [Research hypothesis] (RQ9).

9 Roadmap and Open Decisions

[Direction, 15 Oct 2026 (v0.5r4)] The program runs three tracks. Protocol progress does not establish PMF, and customer interest does not establish protocol correctness. Every milestone names its track.

Track	Priority	Next work	Gate before it moves on
1. PMF experiments	Immediate	Choose buyer and workflow; interviews; synthetic walkthroughs; historical-record analysis with permission	Preregistered continue/narrow/change/stop; evidence of willingness to pay
2. Product viability	Next	Taxonomy and schema; labeled set; thin dashboard; baseline comparison; metrics of Section 7	Preregistered criteria on held-out, customer- and time-split data
3. Protocol development	Continuing	Z01, P01/P02 and the other existing obligations of Papers I–III	Existing acceptance gates, unchanged

Table 2: Three-track roadmap. Outreach, deployment, moving money and training spend each keep their own gate; this paper authorizes none of them.

Pay-for-result settlement (Paper III) is one candidate audience and experiment within track 1. Private payouts (Paper II) remain a gated candidate. Accounting-suite and medical-billing concepts are separate discovery projects.

9.1 Open decisions

1. **First buyer and workflow**, from the candidates of Section 3.2.
2. **Initial jurisdictions**: one or two, chosen with the buyer.
3. **Taxonomy**: purpose categories, mechanics fields, evidence-completeness fields; who defines and validates them.
4. **Data and permissions**: which records, from whom, under what agreement, retention and residency; synthetic or historical first.
5. **Baselines**: rules; Jev or Clef; a general model; whether data may go to a third-party API.
6. **Evaluation criteria**: preregistered thresholds, agreed before each experiment.
7. **Rule ownership**: in-house, a licensed partner, or the customer's advisers.

8. **Reviewer workflow**: who reviews, what they see, how corrections become candidate labels, and label quality control.
9. **Pay-for-result and private payouts**: run in parallel as smaller experiments, or pause until the first buyer is chosen.
10. **Model hosting and privacy posture**: API, self-hosted, or the customer's own infrastructure.

10 Conclusion

ZETTL keeps its broad settlement vision while prioritizing product-market fit and viability. It does so through two proposed efforts: a compliance and analytics dashboard, and a specialized payment decision model, Zlef, designed to be calibrated and able to say it does not know. Together they could support future accounting and tax applications. This paper defines their architecture, trust boundary, model contract and evaluation, and it states plainly what is unknown: the buyer, the data, the accuracy and the value. The next step is learning, not building at scale: one buyer, one workflow, preregistered decisions.

References

[1] Glenn W. Brier. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

[2] Michelle Chen, Alex Reneau, and Kevin Flansburg. Blog post announcing the clef decision models. Cloudflare Blog, 2026. URL https://blog.cloudflare.com/clef-decision-models. 1 October 2026; accessed 2026-10-15.

[3] C. K. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970.

[4] Charles Elkan. The foundations of cost-sensitive learning. In *Proceedings of the 17th International Joint Conference on Artificial Intelligence (IJCAI)*, 2001.

[5] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A survey on concept drift adaptation. *ACM Computing Surveys*, 46(4), 2014.

[6] Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems 30 (NeurIPS)*, 2017.

[7] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*, 2017.

[8] Shachar Kaufman, Saharon Rosset, Claudia Perlich, and Ori Stitelman. Leakage in data mining: Formulation, detection, and avoidance. *ACM Transactions on Knowledge Discovery from Data*, 6(4), 2012.

[9] TypeSafe AI. Introducing system one models & Jev, 2026. URL https://typesafe.ai/blog/introducing-system-one-models-and-jev. 15 September 2026; accessed 2026-10-15.

A Claim and Status Matrix

Claim	Status
Any ZETTL software is running, or any product is live	[Unsupported, needs evidence]: not true; software is a scaffold
UF settlement: a certificate means every validator signed; recovery preserves possibly-certified commitments	[Accepted v0.4] design contract (reviewed argument, bounded model check); not implemented
Privacy levels L0–L3 with user-held disclosure	[Decided v0.5 direction]; mechanics [Proposed, pending Z01]
Post-quantum pool construction	[Proposed, pending Z01], with open obligations
Pay-for-result escrow (phase 0) and portable receipts	[Product, off-protocol]; one candidate experiment
Compliance, regulatory-flags and analytics dashboard	[Product, off-protocol]
Typed payment decision model (Zlef), designed to be calibrated	[Product, off-protocol]; that it beats baselines is a [Research hypothesis]
RLCD-style optimization for ZETTL	[Research hypothesis]; only after measured gaps; not implemented
Dashboard and model create a moat or network effect	[Research hypothesis]
A buyer exists and will pay	[Unsupported, needs evidence]
Accuracy, calibration, savings, cost or latency of Zlef	[Unsupported, needs evidence]: none measured
Jurisdictional coverage, legal certainty, automated tax correctness	[Unsupported, needs evidence]: not claimed
Training-data access, customer commitments	[Unsupported, needs evidence]: none
Jev, Clef and Workers AI availability, pricing and terms	[External dependency]
Jev and Clef benchmark results	[External dependency], vendor-reported; not evidence for this domain
Jurisdiction rules and regulatory texts	[External dependency], versioned and source-cited