KonnectAI Trader

A Systematic Framework for LLM-Augmented Cryptocurrency Trading — Implementation Paper

by **Kai Zeh**

v2.1 — May 2026

KonnectAI Trader · Whitepaper Series

Implementation Paper

Contents

0.1 Abstract / Executive Summary

Retail cryptocurrency algorithmic trading is, by a wide margin, a losing proposition. Independent surveys place the share of retail algo-traders who lose money at approximately 92 %.[1] A 2024 NBER working paper quantifies the principal structural cause: execution slippage — the gap between the price a model sees and the price an order actually fills at — accounts for 34–67 % of the realised underperformance relative to backtests.[2] The signal is rarely the problem. Execution, regime blindness, and the absence of cross-signal context are.

KonnectAI Trader is a systematic framework that addresses these three failure modes directly. Rather than treat a cryptocurrency as a univariate price series and train another neural network on it — a well-documented path to overfitting[3] — we treat each trade decision as a structured analytic problem and delegate it to a large language model (LLM) whose context window has been explicitly engineered to include (i) cross-timeframe technical indicators, (ii) a deterministic market-regime classification, (iii) on-chain flow and network-health metrics, and (iv) a risk posture parameterised by an operator-configurable strategy mode. A rule-based screener filters the candidate universe before any LLM call, which both enforces a quality gate and keeps inference cost bounded. Every LLM output passes through deterministic risk validators — confidence gate and minimum risk/reward ratio — before it can open a live or paper position. Every decision, with its full reasoning and full capital-event context, is persisted to an auditable store.

[1] Kalena Research. *"Regime-Aware Crypto Algorithmic Trading: A 2026 Practitioner Review."* March 2026.
[2] National Bureau of Economic Research. Working Paper 31890. *"Slippage and Performance in Algorithmic Trading."* 2024.
[3] Viprasol Research. *"Algorithmic Crypto Trading in 2026: State of the Art."* Annual review, 2026.

This is not a speculative proposal. **The system has been running live, with Kai Zeh's own capital, since January 2026, with the current feature set in continuous production since April 2026.** The methodology mirrors the empirical pattern that NUS Singapore's CryptoTrade paper (arXiv 2407.09546[4]) and the FinRL-DeepSeek 2025 contest[5] have validated: an LLM given structured, multi-source context with a reflective evaluation loop outperforms classical time-series baselines and buy-and-hold across multiple coins and multiple market phases. We extend that pattern with (i) bidirectional long-and-short execution on Binance Futures, (ii) regime-change-exit logic that closes positions when the underlying regime flips against the trade, (iii) wallet-based net-PnL accounting that survives deposits and withdrawals, and (iv) capital-segregated multi-account architecture suitable for institutional pooling.

What we ask investors to provide. KonnectAI Trader has reached the end of the build-out phase. We are not raising capital to hire a team; we are opening a small, hard-capped pool of trading-capital slots on the operating system. The model is **four investor slots × CHF 100 k = CHF 400 k of total trading capital** per server cluster. Each slot is structured in two tranches: **CHF 50 k at contract signature**, and a **second CHF 50 k tranche due one month after the production hardware is provisioned and live trading is functional under contract terms**. The second tranche may be secured via escrow or bank guarantee at the investor's option. Operating costs are shared at **CHF 6,250/month per investor (CHF 25,000/month total, CHF 300 k/year)**, auto-routed monthly via Binance Pay from the investor's own Binance account to the KonnectAI operator wallet under an investor-controlled settings toggle (default ON).

What investors receive. KonnectAI Trader is explicitly **not a subscription product, not a signals service, not a fund, and not a custodial service**. It is a **non-custodial capital-access platform**. Each investor maintains their **own Binance account**, fully owned and controlled by them at all times, and issues KonnectAI a **scoped API key (spot trade, futures trade, internal transfer for the auto-pay flow — withdrawal permanently disabled, IP-whitelisted)**. The operating logic — the LLM analyser, the regime classifier, the screener, the validators — is shared (one analysis per cycle, fan-out via the four investor API keys, position-sized per individual account equity); the capital itself stays in each investor's directly owned account. The investor retains the legal ownership of all assets and can revoke API access at any moment via the Binance UI.

Performance projections. We project three regime-conditional daily ROI scenarios on the basis of MVP data accumulated since January 2026. We frame these as projections, not as a track record:

[4] Li, Z. et al. (National University of Singapore). *"CryptoTrade: A Reflective-LLM-Agent Framework for Cryptocurrency Trading."* arXiv:2407.09546, 2025.

[5] *"FinRL-DeepSeek Contest: Reinforcement Learning with LLM-Derived Sentiment Features for Financial Decision Tasks."* Contest report, 2025.

Market regime	Daily ROI projection	Annualised (linear, 365 days)
Drawdown	0.35 %	+127 % p.a.
Sideways	0.50 %	+180 % p.a.
Bullish	0.65 %	+237 % p.a.

These figures sit materially above the 15–30 % p.a. envelope reported in the academic literature for LLM-augmented systems.[6] The differentiator is operational, not theoretical: (i) higher-frequency execution (1,440 LLM calls per day on the production cron versus daily/4-hour cycles in published studies), (ii) bidirectional long-and-short execution on Binance Futures (academic baselines are typically long-only), (iii) regime-aware execution including a regime-change-exit branch absent from published designs, (iv) adaptive prompt-and-threshold tuning on live data, and (v) wallet-based capital-event-aware accounting that captures the realised return on the actual deployed capital, not on a paper-simulation.

Defensibility. Our edge is not a secret indicator threshold. It is the composition: rule-based screening + regime classification + on-chain integration + disciplined LLM analysis + bidirectional execution + regime-change exit + deterministic validators + capital-event-aware audit. Each component is replaceable, none is the whole system, and the integration itself is the intellectual property. Author Kai Zeh brings 30+ years of system and platform architecture experience, including a prior CTO/CEO role in which he architected a platform valued by BDO at over CHF 400 million (Section 14).

Capital denomination. Trading capital is held in **USDT** on Binance. CHF amounts shown throughout this document — CHF 100 k slot size, CHF 6,250/month operating-cost share, CHF-denominated performance projections — are **reference values for Swiss-resident investors**; actual position sizes are computed in USDT against the prevailing CHF/USDT spot rate. Investors may deposit any amount they choose into their Binance account; the CHF 100 k figures are example deployments and are not contractual minimum balances within the account.

The remainder of this document describes the theoretical foundations (Section 2), conceptual architecture (Section 3), the analyser and its validators (Section 4), regime detection and the regime-change-exit branch (Section 5), on-chain integration (Section 6), strategy modes and position sizing (Section 7), the role of paper trading as an MVP-only research bench (Section 8), evaluation and KPIs including wallet-based net return (Section 9), operator experience (Section 10), operations including the pooled multi-investor architecture (Section 11), security posture (Section 12), the continuous-improvement and production-migration roadmap (Section 13), the four-slot capital-access investment model (Section 14), the MVP track record (Section 15), references (Section 16), and **Appendix A** — a concrete reference-implementation walkthrough of the live system: technology stack, architectural

[6] Li, Z. et al. (National University of Singapore). *"CryptoTrade: A Reflective-LLM-Agent Framework for Cryptocurrency Trading."* arXiv:2407.09546, 2025.

decision records, code-level descriptions of the capital-events pipeline, the regime-exit helper, the bridge-fix for the equity curve, the wallet-return computation, and the four-investor non-custodial fan-out architecture.

0.2 1. Introduction & Motivation

0.2.1 1.1 The retail algo-trading problem

The dominant narrative around retail cryptocurrency trading in 2020–2024 was that algorithmic tools and backtesting platforms would democratise quantitative strategies. The empirical record has not vindicated that narrative. Kalena Research's March 2026 field survey estimates that approximately 92 % of retail algorithmic traders lose money on a twelve-month look-back.[7] This is consistent with older industry data on retail equity day-trading (Barber & Odean[8]) but is, if anything, more severe in crypto because of three compounding factors:

1. **Thinner order books per venue**, especially outside BTC/ETH perpetual markets. What looks like a liquid market in aggregate is fragmented across dozens of venues, with actual queue depth at any given tick often an order of magnitude smaller than implied by the displayed volume.
2. **24/7 market hours**, which punish inattentive discretionary overrides and reward systems that either run continuously or do not trade at all.
3. **Regime changes that rotate violently**, often in 48–96 hours, between trending and range conditions — horizons on which classical static-parameter systems do not adapt.

The 2024 NBER working paper w31890[9] quantifies the principal mechanical reason for retail algo underperformance: **execution slippage**. The paper disaggregates the backtest-to-live performance gap and finds that 34–67 % of it — depending on venue and order size — is attributable to the difference between the price a signal assumes and the price that is actually filled, net of book-depth, queue priority, and adverse fill on news. Signal logic itself is a secondary factor.

This is a crucial diagnostic. It implies that a system which simply automates a known signal, without incorporating execution realism, is systematically over-optimistic about its live performance. It also implies that improvements in signal quality beyond a certain point produce diminishing returns unless execution is addressed in parallel — and that a system which has been *measured live, on real capital, including fees and slippage*, has a structural credibility

[7] Kalena Research. *"Regime-Aware Crypto Algorithmic Trading: A 2026 Practitioner Review."* March 2026.

[8] Barber, B. M., and Odean, T. *"Trading Is Hazardous to Your Wealth: The Common Stock Investment Performance of Individual Investors."* Journal of Finance, 55(2): 773–806, 2000.

[9] National Bureau of Economic Research. Working Paper 31890. *"Slippage and Performance in Algorithmic Trading."* 2024.

advantage over systems whose performance is reported from backtests or paper-trading only.

0.2.2 1.2 Why LLMs change the calculus

The 2020–2023 generation of retail trading tooling focused on technical indicators and, increasingly, on machine-learning models trained directly on price histories. Viprasol's 2026 state-of-the-art review concludes that **neural networks trained directly on price series remain an overfitting trap**: they memorise, they do not generalise, and they fail catastrophically at regime boundaries.[10] Where ML has succeeded is in narrower tasks — volatility-spike prediction, cross-asset correlation under stress, and execution optimisation — not in direct return forecasting.

What changed the calculus is the emergence of general-purpose LLMs capable of *structured reasoning over heterogeneous inputs*. NUS Singapore's CryptoTrade paper (arXiv 2407.09546, 2025) demonstrates empirically that an LLM agent, given (i) on-chain metrics, (ii) off-chain news, and (iii) a reflective-loop evaluation pattern, outperforms time-series baselines and buy-and-hold across BTC, ETH, SOL and several altcoins over multiple phases of the 2022–2024 market.[11] The 2025 FinRL-DeepSeek contest generalises the pattern further by formalising LLM-derived sentiment as a feature in reinforcement-learning-based execution agents.[12]

The insight is that the LLM is not a "better indicator". It is a different class of component — an *analyst* that can take heterogeneous, partially redundant inputs and produce a disciplined decision with explicit reasoning. Three properties matter:

- **Input heterogeneity.** An LLM can consume RSI on a 4-hour candle, exchange-outflow in BTC terms, and a regime classification, in the same prompt, without a hand-crafted feature engineering step for each.
- **Reasoning artefact.** The LLM produces a natural-language rationale alongside its structured output. This rationale is auditable, reviewable by the operator, and invaluable for post-hoc diagnosis.
- **Structured output under schema constraint.** Modern LLMs, when constrained by an explicit output schema (action, confidence, entry, stop, take-profit levels, reasoning), produce decisions that can feed directly into deterministic risk validators and execution logic.

[10] Viprasol Research. *"Algorithmic Crypto Trading in 2026: State of the Art."* Annual review, 2026.

[11] Li, Z. et al. (National University of Singapore). *"CryptoTrade: A Reflective-LLM-Agent Framework for Cryptocurrency Trading."* arXiv:2407.09546, 2025.

[12] *"FinRL-DeepSeek Contest: Reinforcement Learning with LLM-Derived Sentiment Features for Financial Decision Tasks."* Contest report, 2025.

0.2.3 1.3 What we do differently

KonnectAI Trader is an exercise in *integration discipline* rather than novel algorithms. Every component we use — technical indicators, regime classification, on-chain metrics, LLM-based reasoning, risk validators — is individually well understood. The contribution is the composition, and specifically:

1. **A screening layer before the LLM.** We do not ask the LLM to evaluate every coin every cycle. A rule-based screener identifies a small number of candidates per cycle based on multi-trigger criteria. This is both a cost gate (LLM inference is not free) and a quality gate (the LLM sees only situations that merit analysis).
2. **Structured context, not free-form prompts.** The LLM receives a deterministic prompt bundle with a fixed schema — symbol context, cross-timeframe indicators, regime classification, on-chain block, strategy-mode block — and produces a deterministic JSON response with a fixed schema. This is engineering, not prompting.
3. **Deterministic validators after the LLM.** The LLM can be wrong. A confidence gate and a risk/reward validator are applied to every output, and can downgrade or veto the decision.
4. **Bidirectional execution from day one.** The system trades both long and short on Binance Futures, with native long-or-short signal output from the analyser. Continuity of performance through bull, bear and sideways phases is structural, not aspirational.
5. **Regime-change exit.** When the regime classifier flips against an open trade and the trade is already in profit beyond the round-trip-fee threshold, the position is closed automatically. This branch is, to our knowledge, absent from published academic implementations.
6. **Wallet-based capital-event-aware accounting.** Every deposit, withdrawal, and Spot↔Futures internal transfer is reconciled into the equity curve via geometric period-linking. Net-PnL is reported as a percentage of *deployed capital*, not as a paper-simulation figure.
7. **A complete audit trail.** Every candidate, every analysis, every trade, every parameter change, every capital event is persisted with a timestamp and — in the case of parameter changes — an attribution to the user who made the change.

This is not a novel trading idea. It is a disciplined engineering approach to a problem domain where most participants are underdisciplined.

0.3 2. Theoretical Foundations

0.3.1 2.1 Technical analysis indicators

A rigorous system must treat technical indicators as what they are: summary statistics on price and volume. They are not predictive in isolation; they are useful as inputs to a broader

decision. We briefly recap the indicators we use and the semantics that matter for the LLM prompt.

Relative Strength Index (RSI). Welles Wilder's 1978 formulation[13] measures the ratio of average gains to average losses over a rolling window (typically 14 periods):

$$\text{RSI} = 100 - \frac{100}{1 + RS}, \quad RS = \frac{\overline{\text{gain}}_{14}}{\overline{\text{loss}}_{14}}$$

Values below a lower threshold (conventionally 30, often tuned lower in high-volatility assets) are read as oversold; values above an upper threshold (conventionally 70) as overbought. The critical caveat — frequently ignored — is that in a strong trend, RSI can remain in the "overbought" or "oversold" zone for extended periods without a reversal. RSI alone is not an entry signal; it is a context cue.

Moving Average Convergence/Divergence (MACD). The difference of a 12-period and 26-period exponential moving average, smoothed by a 9-period EMA of that difference. A "flip" — MACD line crossing its signal line — is interpreted as a short-term momentum shift. Its principal failure mode is in chop, where it generates a high rate of false flips.

Bollinger Bands. A moving average flanked by bands at ±2 standard deviations. Price touching the upper band signals stretched upside; the lower band, stretched downside. In a trending regime, price can "walk the band" — that is, ride the upper band upward for many candles without a mean-reversion event. Our screening logic explicitly respects this by never firing a double-sided Bollinger signal within the same cycle on the same symbol.

Moving averages (EMA 20 / 50 / 200 and SMA 20 / 50). The principal use is as a regime filter. Price above the 200-EMA on a daily timeframe is a coarse but robust long-term trend indicator. Crosses between shorter-horizon EMAs (e.g., 20/50) are micro-trend signals.

Average True Range (ATR). An absolute-terms volatility measure. We use ATR, normalised to current price, as the input to our regime classifier and as a reference for volatility-adjusted stop distances.

Volume. Raw traded volume is noisy; the ratio of current volume to a short-horizon trailing average is a cleaner signal of unusual activity. We use a *completed-candle* baseline to avoid contamination from the in-progress candle, which is a subtle but consequential source of bias in many open-source regime classifiers.

The combined use of these indicators — rather than reliance on any one — is what academic meta-studies of technical analysis consistently recommend.[14] The LLM is the component that weighs them against each other.

[13] Wilder, J. Welles. *New Concepts in Technical Trading Systems.* Trend Research, 1978.

[14] Park, C.-H., and Irwin, S. H. *"What Do We Know About the Profitability of Technical Analysis?"* Journal of Economic Surveys, 21(4): 786–826, 2007.

0.3.2 2.2 Market regime theory

A market regime is a latent state that shapes the conditional distribution of price movements. In the Kalena 2026 framework that we adopt as our starting point, five regimes are named:

1. **Trend-up, liquid.** Clear upward slope on short-horizon EMAs, ATR within normal bounds, volume confirming.
2. **Trend-down, liquid.** Symmetric: clear downward slope, ATR normal, volume confirming.
3. **Range.** EMA slopes flat, price oscillating within a narrow channel, volume average or below.
4. **Volatility expansion.** ATR sharply elevated, no clear directional slope, typically preceding or following a news event.
5. **Crash.** Violent downside with elevated ATR and weakening structure (e.g., price below 200-EMA, RSI deeply negative, liquidity thinning).

The critical insight of the regime literature[15][16] is that *the same signal has different expected value in different regimes*. A Bollinger-band touch is mean-reverting in a range regime and continuation-favouring in a trend regime. A volume spike in a vol-expansion regime is a continuation signal; in a range regime it is frequently a fake-out. A well-designed system must therefore either (i) classify the regime and adjust its interpretation of signals, or (ii) use a model — an LLM — capable of this kind of conditional reasoning when the regime is explicitly provided in its context.

We take approach (ii), but with a deterministic regime classifier as an input rather than expecting the LLM to derive the regime from raw data in the same call. This separates concerns and makes the regime classification auditable and backtestable on its own. We extend the pattern with a **regime-change-exit branch** (Section 5.5): when the classifier flips against an open position and the position is already in profit beyond the round-trip-fee threshold, the trade is closed regardless of stop and take-profit levels.

Kalena's own case-study data[17] report that a regime-switching strategy (Sharpe > 1.5 in the study window) materially outperforms a regime-agnostic version of the same strategy on the same data. We do not claim our implementation reproduces that precise Sharpe figure; we do claim that regime awareness is the structural improvement that the literature most consistently identifies, and that regime-change-exit is a natural extension of that principle into the execution layer.

[15] Kalena Research. *"Regime-Aware Crypto Algorithmic Trading: A 2026 Practitioner Review."* March 2026.
[16] Perfumo, Thomas (Kraken Chief Economist). *"Why This Cycle Isn't Like the Others."* Kraken Research, February 2026.
[17] Kalena Research. *"Regime-Aware Crypto Algorithmic Trading: A 2026 Practitioner Review."* March 2026.

0.3.3 2.3 On-chain signals

On-chain metrics are the cryptocurrency equivalent of macroeconomic indicators: slower-moving, structural, and unavailable in traditional asset classes. They are the domain in which firms like Glassnode, Nansen and Arkham Intelligence have built commercial practices. The categories we rely on are:

Exchange flows. Inflows to exchanges typically precede selling; outflows typically precede or accompany accumulation. Glassnode's published research practices treat 7-day rolling net-flow deltas as a leading indicator for short-to-medium-term directional bias.[18]

Wallet concentration and whale tracking. Large holders' behaviour (top-100 and exchange-cluster wallets) is a non-trivial fraction of the directional variance. Methodologically, we identify custody wallets by labelled on-chain addresses (Etherscan labels in the free tier; Arkham or Nansen label databases in the paid tier) and aggregate by cluster, tracking 24-hour and 7-day deltas.

Network health. Bitcoin hash rate, mempool depth, fee markets, active addresses. These are slow-moving structural signals — not useful for intra-minute entry timing, but strongly useful for framing the broader market phase (accumulation, distribution, capitulation).

Gas price as regime indicator (Ethereum). Ethereum gas price is an aggregate measure of demand for blockspace. Sustained low gas in a bull phase is often anomalous and can signal softening demand; spikes around major events are a continuation confirmation for the asset in question.

0.3.4 2.4 Macro and sentiment

Beyond asset-specific technicals and on-chain, three aggregate indicators matter:

- **DeFi TVL (Total Value Locked).** A proxy for capital committed to the ecosystem across chains. Trended TVL (DefiLlama data) is a longer-horizon posture signal.
- **Fear & Greed Index (Alternative.me).** A composite sentiment index on a 0–100 scale. Extreme readings (< 10 or > 90) are contrarian, not confirmatory.
- **BTC dominance.** BTC market cap as a share of total crypto market cap. Trended dominance movements signal rotation between BTC and altcoins, relevant for portfolio-level decisions but also for per-asset context.
- **Stablecoin supply dynamics.** Growth in aggregate stablecoin supply implies dry powder in the system; contraction implies capital exit.

These macro signals inform our weekly macro report and enter the LLM prompt indirectly through the strategy-mode parameter and the regime classifier; they are not per-trade inputs, because they move too slowly to drive intraday decisions.

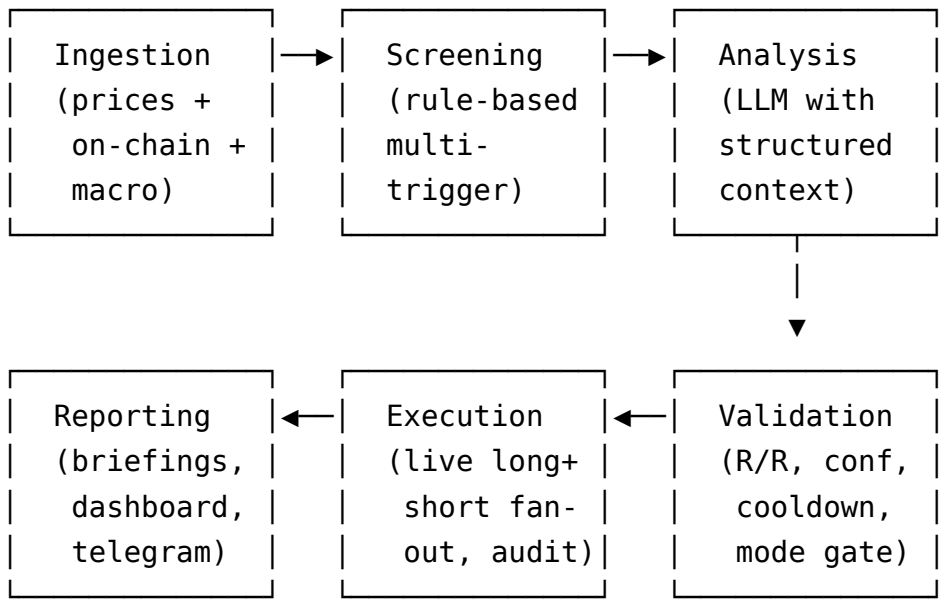
---

[18] Glassnode. *"Week 16/2026 On-Chain Report."* Glassnode Insights, April 2026.

0.4 3. System Architecture (Conceptual)

0.4.1 3.1 Pipeline overview

At the conceptual level, the KonnectAI Trader pipeline is a six-stage flow, run on a fixed cadence:



Each stage is a separate, testable module. The interfaces between stages are strongly typed: the screener emits a candidate record, the analyser consumes candidates and emits analyses, the validator consumes analyses and emits decisions, the executor consumes decisions and emits trades. The reporting layer is a read-only consumer of the persistence tier.

The **cron cadence is configuration**, not architecture. The MVP runs on a 15-minute core loop (96 cycles/day). The production deployment runs on a 1-minute core loop (1,440 cycles/day) using finer-grained candle data, which lifts the upper bound on signal-frequency and is the primary throughput differentiator versus the academic literature. The pipeline shape is the same; only the schedule changes.

0.4.2 3.2 Data ingestion layer

Three sub-streams:

Price / OHLCV / volume. Candle data pulled from Binance public market-data endpoints, at multiple intervals — 1-minute, 5-minute, 15-minute, 1-hour, 4-hour, and 1-day. Indicators are computed server-side on each ingest cycle and stored alongside the raw candles. The ingest cadence on the production cron is synchronised to the 1-minute candle close; on the MVP cron, to the 15-minute candle close.

On-chain. Snapshots of on-chain metrics are fetched hourly from a curated set of sources

(Section 6.1). Each snapshot writes a row with a `(source, asset, metric, value, raw_payload)` tuple. Raw payloads are retained for audit and for retrospective metric derivation. We use a fail-safe pattern: each source is wrapped in its own exception handler; a single source failure does not block the others.

Macro / sentiment. Hourly snapshots of DeFi TVL (DefiLlama), Fear & Greed (Alternative.me), BTC dominance (CoinGecko), and global market cap. These roll up into the weekly macro report and into the regime-classification inputs.

All three streams share a common persistence pattern: append-only with a captured-at timestamp, plus optional derived columns for commonly queried metrics. The raw payload is kept for every row.

0.4.3 3.3 Screening layer

On each cycle, the screener evaluates each coin in the active watchlist against a set of multi-trigger rules. Triggers include RSI threshold crossings, Bollinger-band proximity, EMA crosses (20/50), MACD signal-line flips, and volume spikes. A candidate is emitted when at least N triggers fire on the same symbol within the same cycle (N being a strategy-mode parameter).

The screener serves two purposes:

1. **Cost efficiency.** LLM inference is the single largest variable cost in our pipeline. A screener that reduces the candidate count from the watchlist coins to a small single-digit number of actual candidates per cycle saves an order of magnitude of inference spend.
2. **Quality gate.** Even if inference were free, we would not want the LLM to analyse every coin every cycle. The LLM's marginal value comes from analysing *interesting* situations. Interesting is rule-defined at the screening layer.

A deliberate engineering choice: the screener is entirely deterministic and rule-based. It contains no ML. This makes it reproducible, auditable, and fast to iterate. The LLM enters the pipeline only downstream.

0.4.4 3.4 Analysis layer (LLM-based)

The analysis layer is the heart of the system. For each candidate, the analyser constructs a structured prompt (Section 4.2) that includes:

- Symbol context (current price, 24-hour change, market-cap rank).
- Indicator snapshot across multiple timeframes.
- Recent candle history.
- Market regime classification (Section 5).
- On-chain context block (Section 6) — asset-specific where available, aggregate otherwise.

- Strategy-mode block (Section 7) — the current operator-selected posture.
- Output-schema instruction — a deterministic JSON schema the LLM must populate.

The LLM returns a structured response: `action ∈ {BUY, SELL, HOLD, AVOID}`, a `confidence` score, an `entry` level, a `stop` level, one or more take-profit levels, a `reasoning` narrative, a `risk_level` categorical, and a `timeframe` categorical. The same schema is used for both long and short directions; a SELL action on Binance Futures opens a short position rather than closing an existing long.

Why an LLM? Three reasons, beyond the academic validation cited in Section 1:

- *Compositional reasoning.* A classical scorer would need explicit feature-interaction logic for each signal combination. An LLM produces reasoned output from heterogeneous inputs with a single prompt.
- *Natural-language rationale.* Every trade is accompanied by an English rationale that can be audited, reviewed, and mined for diagnostic patterns. Classical systems produce scores; LLMs produce arguments.
- *Adaptability.* Changing a strategy mode or adding a new signal (e.g., a new on-chain metric) is a prompt edit and a context-bundle update, not a model retrain.

Why Opus-class reasoning specifically. The cost/quality frontier in 2026 has moved decisively toward reasoning-optimised models. Smaller/cheaper models are tempting on unit-cost grounds but underperform materially on multi-signal financial decision tasks in our internal comparisons.[19] We currently use Anthropic's Opus 4-7 family but the framework is model-agnostic; substitution requires only schema-level adaptation.

Pooling: one analysis, many executions. A single LLM call per cycle produces a single structured signal per candidate. The signal is then fanned out via the four investor API keys onto the four investor-owned Binance accounts, sized against each individual account's available equity. This is the core economic property of the pooled architecture (Section 11.4): inference cost scales with cycle frequency, not with the number of investors.

0.4.5 3.5 Risk management layer

Three deterministic filters are applied to every LLM output before it can become a trade:

1. **Confidence gate.** A minimum confidence threshold, strategy-mode-dependent. Outputs below threshold are demoted to HOLD.
2. **Risk/reward validator.** The ratio `(take-profit − entry) / (entry − stop)` for long trades, and its symmetric form for short trades, must exceed a minimum, strategy-mode-dependent. Outputs below threshold are downgraded to HOLD with a confidence penalty. This validator catches an important LLM failure mode: the model occasionally proposes technically plausible entries with insufficient reward per unit of

[19] Li, Z. et al. (National University of Singapore). *"CryptoTrade: A Reflective-LLM-Agent Framework for Cryptocurrency Trading."* arXiv:2407.09546, 2025.

risk.

3. **Cooldown.** After a stop-out on a given symbol, a cooldown period prevents the system from immediately re-entering the same symbol. This is a guard against the "revenge-trade" pattern.

Beyond these per-trade gates, the risk layer includes position-management logic: **partial take-profit with a runner** (Section 8.2), **TP1-aware position sizing** that computes risk against the *original* stop rather than a TP1-trailed stop, **original-SL preservation** so that a TP1 fill never overwrites the originally chosen risk boundary in the audit record, **regime-change exit** (Section 5.5), **timeout closure** for trades that go neither to target nor stop within a defined horizon, and **daily-cap alerts** to prevent signal-alert fatigue.

0.4.6 3.6 Execution layer

All execution flows through a single execution layer, with **live-trading on Binance Spot and Binance Futures** as the production posture. The system has run live since January 2026 on Kai's own capital and against the same execution layer that will service the four investor-owned Binance accounts in the production deployment, via their respective scoped API keys.

The execution layer addresses both modes:

- **Spot (long-only).** Used for the long-side bucket of the strategy where capital efficiency and the simplicity of full-collateral positions are preferred.
- **Futures (long and short).** Used for both the short side and for capital-efficient long exposure. Leverage, margin, and funding are tracked per trade. Stop and take-profit levels are placed as exchange-side conditional orders.

Production-grade execution mechanics:

- **Intrabar fills are honoured by exchange.** The exchange — not our paper-simulator — is the authority on fills. Stops and take-profits are placed exchange-side as conditional orders; our reconciliation layer records the fills as they arrive over the user-data stream.
- **Partial exits are tracked in place, not cloned.** When TP1 is reached, the `remaining_pct` is reduced, the stop is lifted to the TP1 level, and the trade continues in the same record toward TP2. The *original* stop level is preserved as a separate column for risk-attribution accuracy. This avoids the bookkeeping pathology of parallel clone trades.
- **Manual close is a first-class action.** An operator can close any open trade from the dashboard. The action is logged with the closing price and the user who executed it.
- **Regime-change exit** (Section 5.5) is mechanically wired into the cron loop and operates with the same audit discipline as automatic stop and take-profit fills.
- **Capital-event reconciliation.** Deposits, withdrawals, and Spot↔Futures internal transfers are pulled from Binance and reconciled with the equity curve every cycle,

with phantom-dip prevention via internal-transfer bridge logic (Section 9.4).

The Phase-5 build-out narrative of the previous whitepaper version is now complete: live trading is the operating mode, not a future goal. The frontier from here is *scaling*, not *enabling*.

0.4.7 3.7 Reporting and audit

Four reporting surfaces:

- **Dashboard.** Web UI split into a public surface (sanitised, showcase-appropriate) and a private, authenticated surface (full trade detail, equity curve, capital events, settings). Described in Section 10.
- **Telegram briefings.** Twice daily, morning and evening, at operator-local hours. Content: open trades, recent signals, market context, on-chain anomalies.
- **Weekly macro report.** An LLM-generated synthesis of macro and on-chain data over the preceding week, delivered on a fixed schedule.
- **Audit store.** Every candidate, analysis, trade, parameter change, alert, capital event, and internal transfer is persisted with a full timestamped payload. This is the ground truth for post-hoc evaluation and for any future compliance review.

0.5 4. The Analyser: LLM as Disciplined Analyst

0.5.1 4.1 Prompt engineering principles

The difference between an LLM used well and an LLM used badly on a financial-decision task is almost entirely in the prompt architecture. Three principles guide our design:

Principle 1 — Structured context, not free-form prose. The prompt is a deterministic template populated with deterministic data. There is no natural-language request like "please analyse BTC for me". Every section heading and every data block is in a fixed order and format. This is less expressive but far more reproducible; it also makes prompt-version diffing meaningful.

Principle 2 — Explicit regime and posture context. We provide the regime classification and the strategy mode as explicit, labelled inputs, not as inferences the LLM must derive. This both saves context-window real estate and removes a source of variance in the model's reasoning.

Principle 3 — Deterministic output schema. The LLM is instructed to return a JSON object with a specific schema. Modern LLMs are reliable at this under schema instruction, especially when combined with an example payload in the system prompt. The downstream system consumes only the schema-valid response; non-conforming responses are retried once, then dropped.

0.5.2 4.2 Prompt structure (generalised)

The following is a *generalised* template. Our production prompt contains additional tuning we do not publish here. The structure, however, is representative.

```
SYSTEM:
You are a disciplined crypto-market analyst.
Your output must conform to the JSON schema specified below.
Do not add commentary outside the JSON.
Respect the risk posture specified in the STRATEGY MODE section.
You may recommend BUY (long), SELL (short), HOLD, or AVOID.

USER:
ANALYSIS REQUEST

SYMBOL CONTEXT
  Symbol:             [SYMBOL]
  Current price:      [PRICE]
  24-hour change:     [PCT]
  Market-cap rank:    [RANK]

INDICATOR SNAPSHOT (1m / 15m / 1h / 4h / 1d)
  1m:  RSI=..  EMA20=..  EMA50=..  MACD=..  ATR%=..
  15m: (same fields)
  1h:  (same fields, plus EMA200, BB)
  4h:  (same fields)
  1d:  (same fields)

RECENT CANDLES
  1m  (last 30): ...
  15m (last 10): ...
  1h  (last 10): ...
  4h  (last 10): ...

MARKET REGIME
  Regime:           [one of: trend_up_liquid, trend_down_liquid, range,
                             vol_expansion, crash]
  Confidence:         [0..1]
  Key metrics:        EMA20-slope=..  vol-ratio=..  price-vs-EMA200=..
                      RSI-1h-mean=..  ATR-normalised=..

ON-CHAIN CONTEXT
```

```
  Asset-specific (if available):
    24h exchange net-flow (native):  ..
    24h exchange net-flow (USD):     ..
    Tracked-wallet delta (24h):      ..
  Aggregate:
    BTC hash rate (7d avg, TH/s):    ..
    Mempool depth (MB):              ..
    ETH gas (gwei):                  ..
    DeFi TVL (USD, 24h delta %):     ..

STRATEGY MODE
  Posture:             [conservative | moderate | aggressive]
  Min confidence:      [value]
  Min R/R:             [value]
  Direction allowed:   [long_only | short_only | both]
  Notes:               [posture-specific guidance]

OUTPUT SCHEMA
{
  "action":      "BUY|SELL|HOLD|AVOID",
  "confidence":  integer 0..100,
  "entry":       float,
  "stop":        float,
  "tp1":         float,
  "tp2":         float,
  "reasoning":   string (max 500 chars),
  "risk_level":  "low|medium|high",
  "timeframe":   "short|medium|long"
}
```

Three features deserve emphasis:

- The regime classification is *provided*, not inferred. This is a deliberate architectural choice (Section 5).
- The on-chain block is structured with asset-specific and aggregate sub-sections. Where asset-specific data is unavailable (e.g., for smaller altcoins outside major exchange label-sets), the block gracefully degrades to aggregate-only, with a note.
- The strategy-mode block contains *values* (min confidence, min R/R, allowed direction) rather than just a label, so the LLM can reason about why the posture matters without the operator having to modify the system prompt when parameters change.

0.5.3 4.3 Validators

The LLM is not the final arbiter. Two deterministic validators sit downstream:

Risk/Reward validator. The ratio `(tp1 − entry) / (entry − stop)` for long trades, and `(entry − tp1) / (stop − entry)` for short trades, must exceed a strategy-mode-dependent threshold. Below the threshold, the trade is *not rejected silently* — its action is downgraded to HOLD and its confidence is reduced by a fixed penalty. This preserves the audit record: we know what the LLM wanted to do and why the validator intervened.

Confidence gate. Strategy-mode-dependent minimum confidence. Actions below threshold become HOLD; HOLD actions propagate to analysis storage but do not generate trades or alerts.

These validators serve three functions:

1. **Safety net against LLM overconfidence.** Occasionally, the LLM generates a technically plausible trade setup with insufficient R/R. The validator is the mechanical backstop.
2. **Operator control.** By changing the strategy mode (Section 7), the operator can tighten or loosen the overall posture without changing the prompt.
3. **A/B analysis.** Trades are tagged with the strategy mode in effect at creation time, allowing post-hoc comparison of mode-conditional expectancy.

0.6 5. Market Regime Detection

0.6.1 5.1 Motivation

Regime-awareness is, in our view, the single most consequential architectural improvement possible over a naive algorithmic crypto system. The academic and practitioner consensus is converging on the same conclusion:

- Kalena Research (2026) reports material performance differentials — on the order of tens of basis points per week in the case-study window — between regime-switching and regime-agnostic versions of the same base strategy.[20]
- Perfumo (Kraken, 2026) argues that the 2025–2026 market in particular has been characterised by extended range phases interrupted by brief volatility expansions, a configuration punishing to trend-following systems without regime filtering.[21]
- Viprasol (2026) identifies regime-detection-before-position-sizing as the defining feature of post-2025 professional algorithmic trading.[22]

[20] Kalena Research. *"Regime-Aware Crypto Algorithmic Trading: A 2026 Practitioner Review."* March 2026.
[21] Perfumo, Thomas (Kraken Chief Economist). *"Why This Cycle Isn't Like the Others."* Kraken Research, February 2026.
[22] Viprasol Research. *"Algorithmic Crypto Trading in 2026: State of the Art."* Annual review, 2026.

The methodological challenge is that *true* regimes are latent and unobservable; any classifier is an approximation. The design choice is between statistical approaches (hidden Markov models, change-point detection) and rule-based decision trees. We chose rule-based, for three reasons: (i) transparency for audit and operator review, (ii) fast iteration and deterministic testing, (iii) the inputs themselves (EMA slope, ATR, volume ratio) are already estimates of latent variables, so layering a statistical model on top adds parameterisation without adding much information.

0.6.2 5.2 Classifier inputs

Our classifier operates on the 4-hour candle stream and uses the following inputs, computed per symbol:

- **EMA20 and EMA50 slopes.** The percentage change of the EMA over a 5-candle lookback. Positive-and-consistent signals an upward trend; negative-and-consistent signals a downward trend; near-zero signals range.
- **ATR normalised.** The 14-period ATR expressed as a percentage of the current price. A measure of per-candle volatility independent of asset price level.
- **Volume ratio.** Current-candle volume divided by the mean of the preceding 19 *completed* candles. The "completed" qualifier is important: using the in-progress candle as the numerator against a completed-candle baseline produces a systematic undershoot, one of the more common bugs in open-source regime classifiers.
- **Price-versus-EMA200.** Binary indicator of whether price is above or below the daily EMA200.
- **Bollinger-band position.** The relative position of price within the band range.
- **RSI 1-hour rolling mean.** Smoothed RSI to avoid single-candle noise.

0.6.3 5.3 Decision methodology

The classifier is a short decision tree evaluated in a fixed order of priority:

The ordering matters: we check for crash first (so a vol-expansion that is *also* a crash is classified as crash), then vol-expansion, then range, then trend. The thresholds themselves are strategy-relevant and not published here; the methodology for setting them is (i) backtesting across 2023–2025 data, (ii) regime-forward validation on 2026 data, (iii) ongoing statistical monitoring of classifier stability (regime-change frequency as a diagnostic).

Conservatism bias. Our thresholds are set such that, absent clear evidence, the classifier defaults to *range*. Ranges are the modal regime in 2025–2026 (Perfumo 2026), and range-appropriate strategies (mean reversion, fade-to-band) degrade gracefully in weak trends. Trend-appropriate strategies (breakout continuation) degrade poorly in ranges. This conservatism bias is a deliberate asymmetric-loss choice.

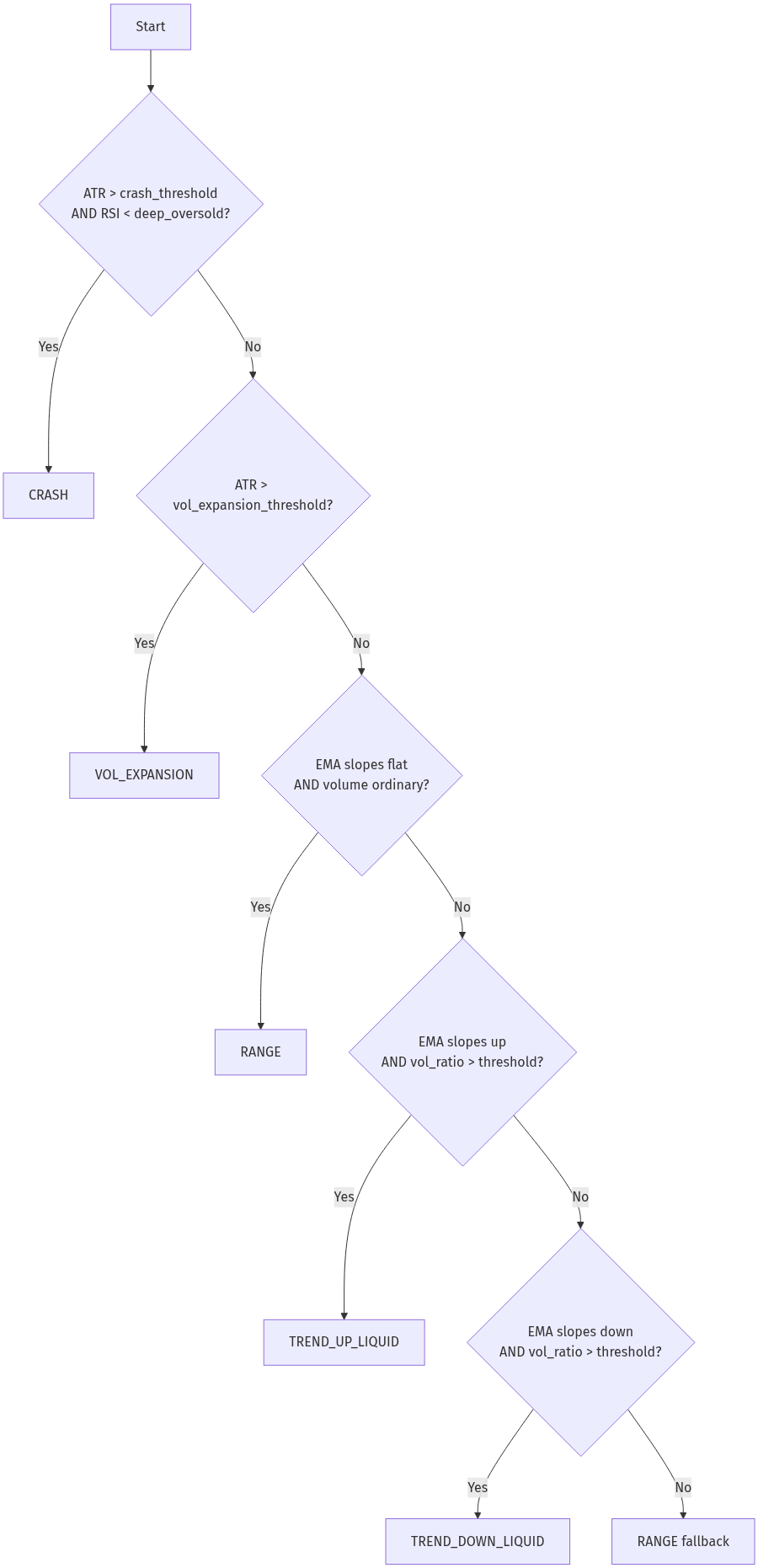


Figure 1: Diagram

0.6.4 5.4 How regime affects the analyser

The regime classification enters the LLM prompt as a labelled input (Section 4.2) and changes how the LLM is expected to reason about the candidate. The prompt does not *instruct* the LLM to adopt a particular strategy; instead, it provides regime as context and trusts that a well-aligned model will produce regime-appropriate outputs. We observe this behaviour in production: the LLM's reasoning text explicitly references the regime ("range regime favours fade toward lower band", "trend-up-liquid supports momentum continuation"), in BUY, SELL, and AVOID calls.

0.6.5 5.5 Regime-change exit

The regime classification is also consumed at the **execution** layer, in the form of a regime-change-exit branch. The logic is:

```
For each open trade:
    direction := bull | bear | sideways  (derived from regime label)
    trade_side := long | short

    if direction is unknown or ambiguous:
        skip (no action)

    elif (trade_side == long  AND direction == bear)
      OR (trade_side == short AND direction == bull):
        # The regime now opposes the trade.

        hypothetical_pnl_pct := pnl_at_current_price / notional
        round_trip_fee_pct   := 2 × taker_fee_pct  (entry + exit)

        if hypothetical_pnl_pct > round_trip_fee_pct:
            close trade at current price
            exit_reason = 'regime_change'
        else:
            # Trade is in profit but the regime change
            # would not cover round-trip fees; let the
            # original stop / TP levels resolve.
            no action.
```

The fee/funding calculation is on **notional** (`qty × price`), which is leverage-agnostic. The regime-exit branch never fires on a losing trade — it never causes a stopped-out position to be closed earlier than its stop level. It only triggers when a profitable position is sitting against a freshly hostile regime, where the expected continuation of the original profit is now negative-EV against a trade in the regime-favoured direction.

This branch is, to our knowledge, absent from published academic implementations of LLM-augmented crypto trading. It is a small piece of code with a meaningful effect on regime-transition periods, where positions established in one regime are exposed to a violently changed expectation distribution.

0.6.6 5.6 Diagnostic: regime distribution

One of the most useful diagnostic outputs of the regime classifier is the empirical distribution of regimes over the watchlist. In a balanced market, we expect most coins most of the time to be in range or mild trend; regime homogeneity across the watchlist (e.g., all coins simultaneously in crash) is a cross-validation signal of market-wide stress. Regime-persistence statistics (how long, on average, does a classification last?) are a diagnostic for classifier stability — a classifier that flips regime every cycle is useless, regardless of in-sample accuracy.

0.7 6. On-Chain Integration

0.7.1 6.1 Data-source landscape

Crypto on-chain data has a tiered cost structure. The landscape relevant to an institutional implementation is:

Free tier (public APIs).

- **Blockchain.com charts endpoints** — network-level BTC metrics, hash rate, active addresses, transaction counts.
- **Mempool.space API** — real-time BTC mempool depth, fee markets, recent blocks.
- **Bitnodes API** — reachable-node counts, network topology coarse signals.
- **DefiLlama API** — DeFi TVL, chain-by-chain breakdowns, stablecoin supply by chain.
- **Coin Metrics Community API** — keyless, rate-limited. Provides exchange-level inflow / outflow metrics for BTC and ETH, and a subset of network metrics.
- **Etherscan public endpoints (V2)** — free tier after key registration, 5 requests/second, 100 k/day. Ethereum-specific: gas, supply, balance lookups against labelled addresses.

Mid-tier (paid, monthly).

- **Glassnode** — Advanced tier (~USD 49/month) for most on-chain indicators; Pro tier (~USD 999/month) for proprietary metrics (True Market Mean, Short-Term Holder Cost Basis, Coin Days Destroyed, realised profit decomposition).
- **Nansen** — Smart-money-wallet tracking, labelled-entity analysis.

Top tier (paid, higher monthly).

- **Arkham Intelligence** — the most comprehensive label database; solves the Coin-

base/Kraken smart-contract-custody tracking problem that is unsolvable with Etherscan labels alone.
- **Chainalysis / TRM Labs** — principally compliance-oriented; less directly useful for signal generation.

Trade-offs. The free tier is sufficient for coarse directional signals and for Bitcoin-specific analysis. The main gap is accurate exchange-custody tracking on Ethereum, where modern exchanges increasingly use smart-contract custody architectures that do not carry the traditional Etherscan labels. Our current free-tier tracking captures approximately 2.66 million ETH across 13 labelled wallets; the true figure for the same exchanges (per Glassnode / Arkham reconciliation) is materially higher. *Relative* movements remain informative; *absolute* levels should not be treated as authoritative until paid-tier tracking is in place. Migration to the Arkham label database is on the post-MVP enhancement track (Section 13).

0.7.2 6.2 Exchange flow as a leading indicator

The interpretative framework — consistent across Glassnode's published research and our own observations — is:

- Sustained **net outflow** from exchanges (coins moving from custodial wallets to self-custody) indicates **accumulation**. Holders are reducing their proximity to a sell venue. This is bullish over a 1–4 week horizon.
- Sustained **net inflow** to exchanges indicates **distribution**. Holders are positioning for sale. This is bearish over the same horizon.
- **Short-horizon spikes** (single-day outliers) are noisy and should not be treated as signals in isolation. The useful window is 7-day rolling net flow.

We expose this to the analyser as the 24-hour net flow, the 7-day rolling net flow, and the percentile of the current reading against a trailing distribution. The LLM then weighs this alongside price action and regime.

0.7.3 6.3 Whale-wallet tracking

Methodologically:

1. **Wallet identification.** For each major centralised exchange, we identify a set of labelled on-chain addresses. Free-tier: Etherscan labels plus public disclosures. Paid-tier: Arkham / Nansen cluster labels, which resolve the smart-contract-custody gap.
2. **Aggregation.** Addresses are grouped by exchange cluster. Balance and net flow are summed per cluster.
3. **Delta tracking.** Per-cluster 24-hour and 7-day balance deltas are the signal. Cluster-internal transfers are not net flows and must be excluded — this is non-trivial in the free tier and is a key advantage of paid-tier label databases.
4. **Threshold-based alerts.** Cluster balance movements exceeding a rolling-distribution

percentile trigger alerts. We do not publish the exact percentile threshold.

0.7.4 6.4 Network health

The slow-moving structural signals:

- **Bitcoin hash rate.** Trending hash rate is a long-horizon confidence proxy (miners commit capital). Sudden hash-rate drops correlate with capitulation phases.
- **Mempool depth and fees.** High mempool with elevated fees signals demand for blockspace — typically accompanying market stress or heightened speculative activity.
- **Active addresses.** Weekly average active addresses, trended, is an adoption proxy.

None of these is an entry signal. All are context signals that affect the analyser's framing — in the LLM prompt, they appear in the "aggregate" sub-section of the on-chain block.

0.8 7. Strategy Modes and Position Sizing

0.8.1 7.1 Three-posture switch

KonnectAI Trader exposes a single operator-controlled setting — the **strategy mode** — with three discrete values: **conservative**, **moderate**, **aggressive**. This setting is global; it applies to all coins in the watchlist.

The mode controls, at minimum:

- The minimum confidence threshold for live or paper trades.
- The minimum R/R ratio enforced by the validator.
- The minimum trigger count required at the screener.
- The RSI oversold/overbought thresholds used at the screener.
- The alert-confidence threshold used for Telegram notifications.
- The directionality envelope (long-only, short-only, both) — a strategy-level overlay on top of the per-trade signal.

We do not publish the specific numerical values of these thresholds per mode. The *methodology* for setting them is: (i) inherit from academic defaults or practitioner heuristics (e.g., RSI 30/70) where applicable, (ii) adjust per mode to produce a measurable posture difference, (iii) validate via in-MVP A/B comparison that each mode has a distinct risk/return profile, (iv) tag live trade records with the mode in effect so that ongoing performance is A/B-comparable.

0.8.2 7.2 Why global, not per-symbol

A per-symbol strategy setting is operationally appealing (different coins want different postures) but creates rapid parameter-explosion: 14 coins × 3 modes × 5 parameters = 210 operational knobs. We consciously chose the global switch as the simpler system, with the explicit future intention to move toward regime-conditional per-symbol adjustments once there is enough data to support per-symbol statistical inference.

0.8.3 7.3 Position sizing

Three position-sizing options are implemented and operator-selectable:

- **Fixed fraction of equity.** E.g., 1.5 % of available equity per trade. Simple, robust, loss-insensitive.
- **ATR-normalised sizing.** Position size is chosen such that the distance to the stop, in absolute terms, equals a fixed percentage of equity. More robust to volatility regime.
- **Volatility-budget sizing.** Portfolio-level "heat" is capped: total simultaneous open risk across all positions is constrained to a portfolio-percentage target.

Two refinements in the production sizing logic deserve note:

- **TP1-aware risk computation.** Risk per trade is calculated against the *original* stop (entry → original-SL distance), not the post-TP1 trailed stop. After TP1 the runner's risk has effectively become small or negative, but the *initial* sizing of the trade was correctly based on the full original-SL distance, not on the more-favourable trailed distance.
- **Capital-event-aware base.** The "available equity" used in sizing is the most recent reconciled equity from the wallet-tracking pipeline (Section 9.4), not a stale snapshot. Deposits and withdrawals propagate into the next sizing cycle without manual intervention.

The professional-practice consensus (Kalena 2026, Viprasol 2026) is that some form of volatility-normalised sizing materially outperforms fixed-percentage sizing, particularly across regime transitions.

0.8.4 7.4 A/B analysis and tagging

Every trade is persisted with a `strategy_mode` tag and a `regime_at_entry` tag. This is not cosmetic: it enables rigorous post-hoc analysis of mode-conditional expectancy. After 30+ trades per mode-and-regime cell, we can compute:

- Mode-specific win rate.
- Mode-specific expectancy (average profit per trade, positive or negative).
- Mode-specific profit factor (gross profit / gross loss).
- Mode-specific drawdown series.

Below that sample size, any inference is not statistically meaningful. This is an honest statement that we make publicly.

0.9 8. Paper Trading Methodology

0.9.1 8.1 The role of paper trading: MVP-only sandbox

In production, all trading is live. There is no paper-trading layer in the production deployment that services the four investor accounts. Every signal is executed against Binance Spot or Binance Futures, on each investor's own account via their scoped API key, with real fees, real slippage, real funding rates.

Paper trading exists in the **MVP** as an R&D bench: a sandboxed strategy-validation environment in which prompt revisions, threshold adjustments, regime-classifier tuning, and new signal sources can be exercised end-to-end without risking capital. The MVP paper-trader is the staging surface; the production trading pool is the live surface; promotions from one to the other are reviewed on-bench before being applied to production.

This positioning is a deliberate change from earlier framings. Paper trading is not the validation of a future live system; the live system already exists and produces the only performance data we report on. Paper trading is the development bench that supports continuous improvement of the live system.

0.9.2 8.2 Partial take-profit with runner

The execution pattern, applied to both live and paper trades:

```
Open trade at `entry`, with stop at `original_SL`, tp1 at `tp1`, tp2 at `tp2`.

Phase 1:  remaining_pct = 100
          stop_active = original_SL
        - If price hits stop_active:  close all at stop.    Reason: stop.
          - If price hits tp1:          close 50 % at tp1.
                                        remaining_pct := 50
                              stop_active := tp1  (raise stop to TP1, locking
                                                   in the TP1 gain on the runner)
                                        original_SL preserved unchanged
                                        continue (next cycle).

Phase 2:  remaining_pct = 50
       - If price hits stop_active:  close all at stop_active.  Reason: tp1_trail_sto
        - If price hits tp2:        close all at tp2.          Reason: tp2.
```

Original-SL preservation. A subtle but important property: the *original* stop level is preserved as a column on the trade record even after TP1 has trailed the active stop. Two reasons: (i) risk attribution at sizing time was based on the original stop, and (ii) post-hoc analysis of "what was the maximum adverse excursion against the original-risk premise" is meaningful only if the original stop is still accessible.

Rationale for partial-TP-runner. A full-exit at TP1 captures modest gains and forgoes upside in trending regimes. A no-scaling approach with single TP2 target too rarely triggers: in our observed live data, TP2 is reached without TP1 being reached first in only a small minority of cases, and the trade frequently stops out en route to TP2 when TP1 has already been traversed. The partial-exit-with-runner preserves the bulk of upside while locking in a portion of the profit and removing downside risk on the runner leg.

Timeout. Trades that reach neither stop nor take-profit within a fixed horizon are forcibly closed at the then-current price. The timeout is in place primarily to prevent trades from drifting indefinitely.

Manual close. An operator can close any open trade from the dashboard. The close is recorded with the operator's identity and the closing price. This is a first-class execution action, not a back-door.

Regime-change close. Independently of stop, take-profit, or timeout, a trade may be closed under the regime-change-exit branch (Section 5.5).

0.9.3 8.3 Slippage in paper vs live

Paper trading does not simulate slippage. This is a deliberate scope limitation, flagged explicitly: paper-trading PnL figures are best-case bounds, not live-expectancy forecasts. The live system, by contrast, *measures* slippage trade-by-trade — every exchange fill is reconciled against the signal-time price, and the realised slippage cost is captured in the per-trade PnL and the wallet-based net return.

This is a structural reason to prefer live numbers over paper numbers in any investor-facing performance discussion: the live numbers already include the largest practitioner-measured drag (NBER 2024[23] reports 34–67 % of backtest-to-live performance gap as slippage), while paper numbers do not.

0.9.4 8.4 Sample-size honesty

Statistically meaningful inference requires sample sizes. Our discipline is:

- Below 30 trades per (mode, regime) cell: no inference.
- 30–100 trades: coarse expectancy estimate, wide confidence intervals.
- 100+ trades per cell: narrowing confidence intervals, meaningful A/B.

[23] National Bureau of Economic Research. Working Paper 31890. *"Slippage and Performance in Algorithmic Trading."* 2024.

- 300+ trades per cell, distributed across regimes: defensible expectancy.

Investors should expect that performance reporting reflects this discipline. We do not extrapolate from small samples.

0.10 9. KPIs and Evaluation Methodology

0.10.1 9.1 Per-trade and aggregate KPIs

The KPIs we track are standard in systematic trading literature. Their standardisation is a feature — investor due-diligence consultants can benchmark our figures against comparable systems without re-engineering our metrics.

- **Win rate** — share of closed trades with positive PnL. Read alongside average win and average loss.
- **Expectancy** — `(win_rate × avg_win) − ((1 − win_rate) × avg_loss)`. A single-number summary of per-trade expected PnL.
- **Profit factor** — gross profit divided by gross loss. Values above 1.25 are professionally acceptable; above 2.0 are strong; below 1.0 are loss-making.
- **Sharpe / Sortino ratio** — return-to-volatility and return-to-downside-volatility, annualised. Conceptually well-defined for a system with a regular trade cadence; careful interpretation in crypto because of fat tails.
- **Maximum drawdown** — the peak-to-trough decline on the equity curve.
- **Per-regime performance.** Expectancy conditional on the regime classification at trade entry. Regime-conditional expectancy is the diagnostic that tells us whether the regime classifier is adding value.
- **Per-strategy-mode performance.** Mode-tagged A/B analysis as described in Section 7.4.
- **Per-direction performance.** Long vs short expectancy, conditional on regime, is a structural diagnostic for the bidirectional architecture.

0.10.2 9.2 Cost KPIs

The system's operating cost is itself a first-class KPI:

- **Cost per analysis** (USD). Total LLM inference spend divided by analyses performed.
- **Cost per winning trade** (USD). Total LLM inference spend divided by count of winning closed trades. A useful holistic metric — a system that makes many analyses but few winning trades is expensive whatever its accuracy.
- **Total daily cost** (USD). The operational envelope. On the production 1-minute cron, with the four-investor pooled-execution architecture, this scales with cycle frequency, not with investor count.

0.10.3 9.3 Wallet-based net return

The headline performance KPI is **wallet-based net-PnL %**. It is computed as follows:

1. **Capital events** (deposits, withdrawals) are pulled from Binance and persisted with their occurred-at timestamp and USD-equivalent value.
2. **Internal Spot↔Futures transfers** are pulled and persisted to a separate audit-only table — they are *not* counted as capital events. (Treating internal transfers as deposits or withdrawals would corrupt the equity curve.)
3. The equity curve is partitioned into **periods** by capital-event boundaries: a period is the interval between two consecutive deposits-or-withdrawals.
4. Within each period, the per-period return `r_i` is computed as `(equity_end − equity_start − net_capital_event) / equity_start`.
5. The period returns are then **geometrically chained**: `total_return = (1 + r_1) × (1 + r_2) × ... × (1 + r_n) − 1`.

This methodology produces a return figure that **survives capital events**: depositing CHF 10 k mid-period does not appear as a 10 % "gain" on the equity curve, and withdrawing CHF 10 k does not appear as a 10 % "loss". The reported wallet-based return is the actual return on capital deployed.

Bridge-fix for internal transfers. Internal Spot↔Futures transfers occur asynchronously across two wallets. Naive aggregate-equity computation produces phantom dips (the transferred amount is briefly absent from both wallets' snapshot windows) which contaminate intra-day equity curves. The bridge-logic reconciles these transfers as zero-net events on the aggregate curve, eliminating the phantom dips. This was specifically engineered for the production deployment after observation in MVP.

0.10.4 9.4 Audit trail

Capital events, internal transfers, live trades, futures trades, paper trades (MVP only), analyses, candidates, and regime classifications are persisted in dedicated tables. Each row has:

- Timestamp (creation and, where applicable, update).
- Full payload including inputs, outputs, and reasoning.
- An external-id (where it originates from a Binance event) marked UNIQUE for idempotent re-sync.
- Attribution to the operator (for manual actions) or the service identity (for automated actions).
- Version tag identifying the pipeline version in effect.

The audit store is append-only by convention. Schema evolution is handled via additive migrations; destructive changes require a documented ADR.

0.10.5 9.5 Performance projections — explicit methodology

Any numerical performance projection for an investor-facing document is a claim that must be defensible. We separate two kinds of figure:

- **Live observations from the MVP since January 2026** — descriptive, used internally as the ground truth that motivates the projection methodology.
- **Forward projections** — explicitly framed as projections, anchored both in MVP data and in a regime-conditional model.

We project performance in three regime scenarios:

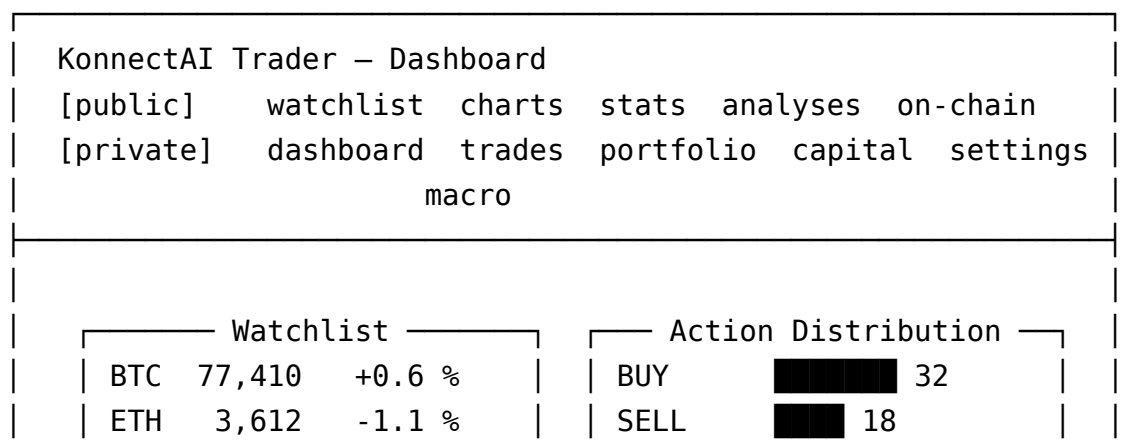
Market regime	Daily ROI projection	Annualised (linear, 365 days)
Drawdown	0.35 %	+127 % p.a.
Sideways	0.50 %	+180 % p.a.
Bullish	0.65 %	+237 % p.a.

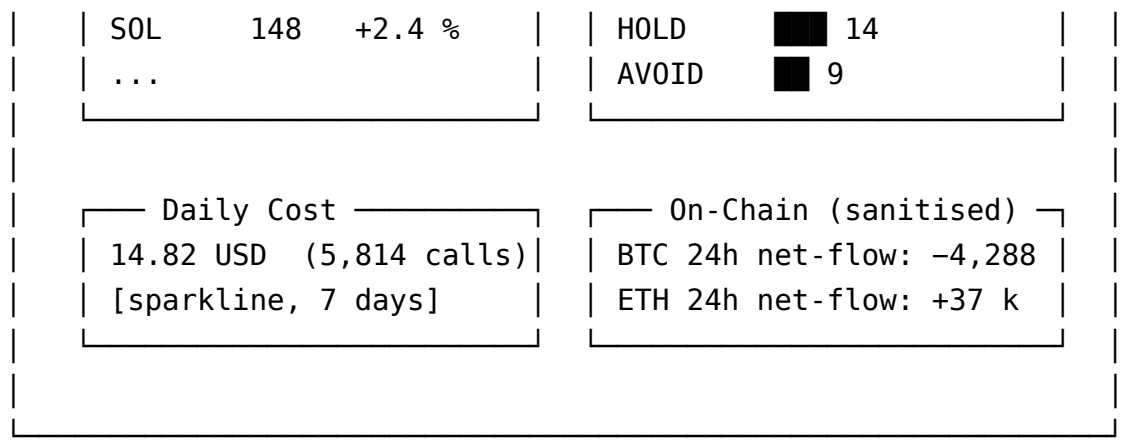
These are **projections on the basis of MVP data**, not historical track records. The lift over the academic 15–30 % envelope is explained by five operational factors: 1,440-cycle cron granularity in production, bidirectional long-and-short execution, regime-aware execution including regime-change exit, adaptive prompt-and-threshold tuning, and capital-event-aware accounting (Section 1.3).

The investor-facing capital-deployment scenarios in Section 14 use these three regime scenarios as the evaluation grid. Investors should focus particularly on the **drawdown scenario** as the worst-case argument: under the most pessimistic assumption, even the most leveraged scenario in the table reaches breakeven within 24 months.

0.11 10. Dashboard and Operator UX

The operator experience is split by authentication boundary. A screenshot-free wireframe of the current design:





0.11.1 10.1 Public surface

The public surface is the demonstrable, no-authentication view. Its purpose is to show the system is live and functional, without exposing trade-level data that would constitute investor-relevant information asymmetry. Content:

- **Watchlist** — current prices, 24-hour percentage change, market-cap rank.
- **Charts** — per-symbol candle chart with indicator overlays (TradingView lightweight-charts in our current implementation).
- **Stats** — analysis-action distribution (counts of BUY/SELL/HOLD/AVOID over rolling windows), aggregate throughput figures.
- **Recent analyses (sanitised)** — each recent analysis shown with action and a redacted reasoning excerpt. Specific entry/stop/TP levels are not public.
- **Daily cost telemetry** — rolling spend tracker, as a transparency and trust signal.
- **On-chain summary** — aggregate network-health, exchange flows at 24-hour granularity, DeFi TVL.

0.11.2 10.2 Private / authenticated surface

Behind authentication:

- **Full analyses** — each analysis with its entry, stop, take-profit, reasoning text, risk level, timeframe, and cost.
- **Open and closed trades** — full live and futures-trade records with entry, exit, PnL, exit reason (stop / tp1 / tp2 / regime_change / timeout / manual), mode tag, regime-at-entry tag, original-SL column, optional notes.
- **Equity curves** — wallet-based net-PnL with capital-event markers, asset-breakdown stacked-area chart, and bridge-fixed Spot/Futures aggregate curve.
- **Portfolio snapshots** — periodic snapshots of holdings on the operator's exchange account, with dust-threshold filtering.
- **Capital events** — chronological view of deposits, withdrawals, and internal transfers with their reconciled USD-equivalent values.

- **Settings** — strategy-mode switcher, watchlist editor, threshold tuning (superuser-only).
- **Macro reports archive** — the weekly LLM-generated macro synthesis, browsable by date.

0.11.3 10.3 Authentication

Authentication uses **TOTP** (time-based one-time password, RFC 6238). We do not use passwords as a primary factor. The rationale:

- Passwords are the dominant breach vector in small-team web applications. Eliminating them entirely eliminates most of the attack surface.
- TOTP secrets are stored server-side, hashed where feasible, and never transmitted after initial provisioning.
- Replay protection is provided by the 30-second TOTP window; a short-lived session token (JWT) is issued after successful TOTP and refreshed via a separate flow.
- Rate limits on the login endpoint prevent TOTP brute-force; standard account-lockout logic applies.

This is appropriate for a small authorised operator set (the founder plus up to four investors plus their authorised principals). Per-investor account views are gated by the user-to-investor-account binding maintained in the auth store, which references the API-key-vault entry rather than holding any direct credentials.

0.12 11. Operations and Observability

0.12.1 11.1 Scheduling

The pipeline is scheduled at multiple cadences. The **core-loop** cadence is configuration:

- **MVP cron: 15-minute core loop** (96 cycles/day). This is the cadence under which the system has run live since January 2026.
- **Production cron: 1-minute core loop** (1,440 cycles/day). This is the target cadence for the four-investor production deployment, exploiting finer-grained candle data and the throughput advantage versus academic baselines.

Other cadences:

- **Hourly on-chain snapshot.** Lower cadence acceptable because on-chain metrics move slowly.
- **Hourly macro snapshot.** Same rationale.
- **Hourly portfolio snapshot.** Read-only Binance API call.
- **Capital-events sync.** Every cycle the deposit / withdrawal / internal-transfer log is reconciled with Binance.

- **Twice-daily briefings.** Morning and evening Telegram briefings at operator-local hours.
- **Weekly macro report.** LLM-generated synthesis on a fixed schedule.

The MVP cadence was chosen as a balance between data freshness and cost during the build-out and validation phase. The production cadence is designed for the high-frequency operating envelope of the four-investor deployment, where the LLM-inference cost is amortised across all four investor accounts (Section 11.4).

0.12.2 11.2 Monitoring

Each scheduled job is monitored for completion and runtime. Missed executions or excessive runtime trigger operator alerts. Key health signals:

- Latest ingest timestamp per data source (staleness detection).
- Analyser call rate and cost envelope (budget-alerting at daily and monthly thresholds).
- Open-trade count per investor account (sanity check against expectations).
- Regime-distribution over the watchlist (cross-validation against market context).
- Screener-emission rate (if it goes to zero, a silent upstream failure is likely).
- Capital-events reconciliation drift (if our reported equity diverges from Binance's wallet snapshot beyond a threshold, alert).
- Internal-transfer bridge integrity (no phantom dips on the aggregate equity curve).

Logs are structured (timestamp, level, module, event, correlation ID) and shipped to a log-aggregation tier appropriate for scale. For the current deployment, the journald-plus-file model is sufficient; at multi-cluster scale, a managed log store (Datadog, Grafana Cloud, or equivalent) is the natural upgrade.

0.12.3 11.3 Audit

Every decision artefact — candidate, analysis, trade, alert, setting change, capital event, internal transfer — is persisted as a row in a structured store with timestamp, full payload, attribution, and pipeline version. The audit store is append-only by convention and append-only-with-additive-migrations by policy. A regulator or investor-appointed auditor can reconstruct the system's behaviour at any point in time.

0.12.4 11.4 Pooled multi-investor architecture

The four-investor operating model is supported by an architecture that **pools the operating logic and keeps capital in investor-owned, non-custodial accounts**. We deliberately do **not** use Binance sub-accounts: that mechanism would require KonnectAI to operate a Binance VIP master account and would put investor funds under our umbrella, which is exactly the custodial posture we refuse to take. Instead:

- **Investor-owned Binance accounts.** Each investor opens (or uses an existing) standard-retail Binance account in their own name, completes Binance's KYC/AML directly with the exchange, and deposits CHF/USD via a Swiss fiat on-ramp (e.g. SEBA, Sygnum, Bitstamp, Kraken Pro, or any regulated route they prefer) which they convert to USDT inside their account.
- **Scoped API keys, withdrawal permanently disabled.** The investor issues KonnectAI a Binance API key with a precisely scoped permission set: spot & margin trading, USDS-M futures trading, and `Universal Transfer` / Binance Pay (required for the monthly operating-cost auto-deduct). **Withdrawal is disabled at all times.** The key is IP-whitelisted to the KonnectAI execution server. The investor can revoke the key in seconds via the Binance UI, which immediately stops all KonnectAI trading on that account.
- **One LLM call per cycle, fan-out via four investor API keys.** The analyser runs once per cycle per candidate. The resulting signal is then executed in parallel via the four investor API keys onto the four investor accounts, with position size on each account computed against that account's available equity. The cost of the analysis is shared four ways: per-investor LLM cost = TotalLLM / 4.
- **Capital-segregated by construction.** Each investor's equity curve, deposits and withdrawals, and PnL are properties of their own Binance account and of the audit records keyed to their API-vault slot. There is no shared capital pool; segregation is a fact about the underlying Binance accounts, not a bookkeeping convention on our side.
- **Operator wallet for cost-routing via Binance Pay.** Operating costs (CHF 6,250-equivalent USDT per investor per month, Section 14) are auto-routed monthly via the Binance Pay endpoint (`/sapi/v1/pay/transactions`) from the investor's account to a dedicated KonnectAI operator wallet. The transfer occurs on the 1st of each month at 09:00 Europe/Zurich. An **investor-controlled settings toggle** governs the auto-deduct (default ON); if the investor turns it off, the system pauses trading on that account until the operating cost has been settled manually. There is no pre-payment and no refund: the model is pay-as-you-go.
- **Hard cap of four investors per cluster.** This is an operational, not arbitrary, limit. Above four investor accounts, the per-account capital share for a given signal becomes too small to be efficient at the round-lot sizes Binance imposes on certain symbols, and the variance in fan-out fill outcomes increases. The four-cap ensures every investor sees substantively the same execution quality.
- **Scaling pathway.** Demand beyond four investors is met by deploying a **second cluster** with the same architecture: separate hardware, separate operator wallet, separate four-investor pool of investor-owned accounts. Each cluster runs its own LLM analyses and its own execution loop. This is parallel scale-out, not vertical fan-out beyond the cap.

The pooled-architecture economics are the key to the cost model: per-investor operating cost is bounded because the largest variable cost (LLM inference) does not scale with in-

vestor count within a cluster. The custodial posture is the key to the regulatory model: at no point do investor funds enter a wallet KonnectAI controls, and at no point can KonnectAI move funds off-exchange (Section 12, Section 14).

0.13 12. Security and Risk Posture

0.13.1 12.1 Secrets management

Production secrets — investor exchange API keys, LLM provider keys, database credentials, webhook tokens — are stored in an **encrypted API-key vault**, a credentials-management tier separate from the application code. At current scale, this is a hardened, OS-permission-locked encrypted store on the execution server; at multi-cluster scale, AWS Secrets Manager or HashiCorp Vault is the natural upgrade path. Investors deliver their API key and secret to KonnectAI through an encrypted channel (PGP, Signal, or equivalent) and the value is ingested directly into the vault.

Three invariants:

- **No secret enters a log.** Logging layers scrub known secret patterns before emission.
- **Least privilege per secret.** Read-only market-data keys are distinct from trade-execution keys. **Every investor-provided trade-execution key has withdrawal permanently disabled and is IP-whitelisted to the KonnectAI execution server**, scoped only to spot trading, futures trading, and the Binance Pay / `Universal Transfer` permission required for the monthly operating-cost auto-deduct (Section 11.4, Section 14).
- **Investor revocability.** The investor retains the ability to revoke their API key at any moment via the Binance UI; revocation is instantaneous and immediately stops all KonnectAI trading on that account.

0.13.2 12.2 Authentication and authorisation

Section 10.3 covers TOTP for the dashboard. For programmatic access:

- LLM provider access uses provider-issued API keys, rotated on a scheduled basis.
- Exchange read-only access (where used internally for market data) uses IP-restricted API keys.
- Each investor's exchange trade access uses an investor-issued, IP-restricted key with withdrawal permanently disabled, scoped to spot, futures, and `Universal Transfer` / Binance Pay only.
- Investor API keys are siloed in the vault; a compromise of one investor's key does not propagate to the others, and in any case the worst-case impact of a compromise is bounded by the no-withdrawal scope: a malicious actor cannot exfiltrate funds, only execute trades on the affected account, on which the investor can also revoke the key

in seconds.

0.13.3 12.3 Rate limits

Inbound API endpoints on the dashboard are rate-limited at the reverse-proxy layer. Outbound API usage respects provider limits (Etherscan 5/s, Coin Metrics Community lightweight, Anthropic tier-appropriate, Binance request-weight scheduling). Backoff-with-jitter is used for transient upstream failures.

0.13.4 12.4 Live-trading safety rails

The live-trading mode operates with the following mechanical guards, in production today:

- **Confirmation buttons in Telegram for every trade above a configurable size.** The operator must click to confirm before the order is submitted on the affected investor account.
- **Max daily loss circuit breaker.** If cumulative realised loss in a calendar day exceeds a threshold, automated trading is disabled until operator re-enablement.
- **Position-sizing guardrail.** Size per trade is bounded by an account-equity-percentage ceiling, regardless of what the LLM proposes.
- **Withdrawal disabled on every investor API key.** No KonnectAI-held credential, on any investor account, can move funds off-exchange. This is enforced at the Binance permission level, not in our application logic, and cannot be re-enabled without the investor explicitly editing their own key in the Binance UI.
- **Cooldown after stop-out.** Per-symbol cooldown prevents revenge-trade re-entry.
- **Regime-change exit only triggers in profit beyond fees.** A regime-change exit cannot turn a winning trade into a loser at execution time.

These are not novel ideas — they are standard institutional-grade safety practice. What matters is that they are built in and have been operating in production since the system went live.

0.14 13. Roadmap

0.14.1 13.1 Completed phases (operational since January 2026)

- **Phase 0 — Foundation.** Database schema, exchange-API client, candle-and-indicator pipeline.
- **Phase 1 — Daily Intelligence.** Screener, LLM analyser, paper-trading sandbox, alerts, briefings.
- **Phase 2 — Strategy Modes.** Settings store, operator-switchable posture, mode tagging on trades.
- **Phase 3 — Dashboard.** TOTP-authenticated web UI, public/private split.

- **Phase 4 — Macro.** Weekly macro-report generator, macro-snapshot persistence.
- **Phase 4.5 — On-chain and regime.** Six free-tier on-chain sources; 5-state regime classifier on the 4-hour candle stream.
- **Phase 5 — Live trading on Spot.** Real capital, real fees, real slippage. Operational since early 2026.
- **Phase 6 — Live trading on Futures (long and short).** Bidirectional execution. Operational.
- **Phase 7 — Capital-event-aware wallet accounting.** Geometric period-linking for net-PnL %, internal-transfer bridge, capital-events sync. Operational since April 2026.
- **Phase 8 — Regime-change exit.** Auto-close on adverse regime flip in profit. Operational since April 2026.
- **Phase 9 — Original-SL preservation, TP1-aware sizing, dust-threshold portfolio view.** Operational since April 2026.

0.14.2 13.2 Continuous improvement (in flight)

- **Production-cron migration.** The MVP runs on a 15-minute core loop. The production deployment for the four-investor cohort migrates to a 1-minute core loop, with finer-grained candle ingest and a higher inference budget per cycle. This is an operational rollout, not a research project.
- **Pooled-execution rollout.** The fan-out across multiple investor API keys has been validated against test accounts paralleling the production execution layer. Onboarding the first investor's own Binance account into the live fan-out is a deployment exercise (vault ingestion + IP whitelist + dry-run signal echo + first live cycle), not a build-out.
- **Adaptive threshold tuning.** Strategy-mode thresholds are reviewed against live data on a fixed cadence. The MVP paper-trader is the validation bench for revisions.
- **Regime-conditional execution refinements.** The regime-change-exit branch is operational. Future refinements include regime-conditional position sizing and regime-conditional take-profit ladders.

0.14.3 13.3 Future enhancements (not blocking the production pool)

- **Paid-tier on-chain integration.** Glassnode Pro, Arkham Intelligence label database for resolving the Coinbase/Kraken smart-contract-custody gap.
- **Social-sentiment integration.** Twitter / Reddit sentiment as an additional signal stream, subject to quality and dedupe discipline. Semantic deduplication of news/social signals to avoid over-weighting echo-chamber narratives.
- **Multi-chain expansion.** Solana, Base, BSC as first-class chains alongside Bitcoin and Ethereum.
- **Multi-cluster scale-out.** Beyond four investors per cluster, additional clusters with the same architecture are deployed in parallel.
- **Compliance and reporting package.** For investors who want to operate the platform

under EU MiFID-II-equivalent or Swiss FINMA-equivalent frameworks.

The framing has changed materially from previous versions of this document: the roadmap is no longer a build-out plan, it is a continuous-improvement and scale-out plan against an operational baseline.

0.15 14. The Investment Model

0.15.1 14.1 What we are offering

KonnectAI Trader has reached the end of the build-out phase. **We are not raising capital to hire a team.** We are opening a small, hard-capped pool of trading-capital slots on the operating system.

Four investor slots × CHF 100 k = CHF 400 k of total trading capital per server cluster.

Each slot is structured in two tranches:

- **Tranche 1: CHF 50 k at contract signature.** Funds the production-cluster hardware procurement and the per-investor onboarding (vault setup, key ingestion, IP whitelist, dry-run validation). The investor's own Binance account is set up by the investor directly with Binance; KonnectAI does not open or hold any account on the investor's behalf.
- **Tranche 2: CHF 50 k, due one month after the production hardware is provisioned and live trading is functional under contract terms.** This tranche is the second half of the trading capital. At the investor's option, it may be secured via escrow or bank guarantee in the intervening period.

Total capital per investor: CHF 100 k. Total funded trading capital per cluster, at full subscription: **CHF 400 k**.

0.15.2 14.2 What investors receive

Per investor:

- **Trading on their own Binance account, fully owned and controlled by them.** KonnectAI never holds custody of the funds. The investor opens the account directly with Binance (a standard retail account is sufficient — no VIP status required), completes Binance's KYC/AML, deposits CHF/USD via a Swiss fiat on-ramp of their choice, and converts to USDT inside the account. The account remains in the investor's name, under their direct legal ownership, at all times.
- **A scoped API integration with KonnectAI.** The investor issues KonnectAI an API key with the following permission profile (see Section 14.7 for the explicit listing): spot & margin trading enabled, USDS-M futures trading enabled, `Universal Transfer` / Binance Pay enabled (for the monthly operating-cost auto-deduct), **withdrawal per-**

manently disabled, IP-restricted to the KonnectAI execution server. The investor can revoke this key at any time via the Binance UI.

- **An equity curve, deposit / withdrawal log, and PnL history** computed against their own account's wallet history. Each investor's view in the dashboard is filtered to their account; segregation is structural, not bookkeeping.
- **Access to the operating system's signals, executed against their account.** Each cycle, the LLM analysis is fanned out via the four investor API keys onto the four investor accounts, sized against each account's available equity.
- **Monthly compounding by default.** Net-PnL stays in the investor's account every cycle; there is no "lock-up" of profits, and the default behaviour is to let returns compound on the trading base. The investor may, in settings, enable an optional monthly auto-distribution of either a fixed percentage or a fixed amount of net-PnL to an external cold wallet of their choice.
- **Quarterly investor reports** with the wallet-based net-PnL %, regime-distribution observed in the period, and a narrative summary.

What the platform is **not**:

- **Not a subscription product.** The investor does not pay a monthly fee for "signals".
- **Not a signals service.** Signals are not delivered as a copyable artefact; they are executed against the investor's account.
- **Not a fund.** There is no fund vehicle, no NAV-calculation methodology, no fund-level fee structure. Each investor is the legal owner of the assets in their own Binance account.
- **Not a custodian.** KonnectAI never holds investor funds. Funds remain in the investor's directly owned Binance account at all times. API permissions are scoped to trading and intra-Binance internal transfers only — withdrawal permissions remain disabled at all times.
- **Not a managed-account agreement under FINMA-licensed asset management.** It is a non-custodial execution service; we are not licensed asset managers and do not represent ourselves as such. The investor retains legal ownership and operational control of their account at all times. This positioning ensures KonnectAI Trader is **not subject to FINMA-licensed asset-management requirements**, because we do not hold custody of investor funds.

0.15.3 14.3 Operating costs

The shared cost model:

- **CHF 6,250-equivalent USDT / month per investor** (CHF 75 k-equivalent / year per investor).
- **Total at full subscription: CHF 25,000-equivalent USDT / month** (CHF 300 k-equivalent / year, the operating envelope of the cluster).

- **Auto-pay mechanics.** On the 1st of each month at 09:00 Europe/Zurich, the operating-cost amount is auto-routed via the Binance Pay endpoint (`/sapi/v1/pay/transactions`) from the investor's account to the dedicated **KonnectAI operator wallet**. The transfer is logged in the audit store with the same `external_id` discipline as any other capital event.
- **Investor-controlled settings toggle.** The auto-pay flow is governed by a per-investor settings toggle (default ON) that the investor controls in their dashboard view. If the investor turns the toggle OFF, the system pauses all trading on that account until the operating-cost amount has been settled manually. This is an **investor-sovereign control feature**, not a compliance workaround: the investor decides whether each month's operating cost is paid, and the system refuses to trade without that authorisation.
- **Pay-as-you-go.** No operating cost is pre-paid. At contract termination, no refund mechanism is required: the investor simply revokes the API key and stops authorising the next month's auto-pay; trading ceases immediately.
- **Net-PnL after costs.** What remains in the investor's account after the monthly auto-pay is the investor's net PnL for the period.

The CHF 6,250/month per-investor figure is the steady-state full-cluster operating cost split four ways. It covers, in aggregate across the cluster:

- LLM inference at the production 1-minute cron cadence (the largest single line item).
- Server hosting, including dedicated production hardware and managed services.
- An operator stipend covering monitoring, on-call response, threshold tuning, prompt revisions, and continuous-improvement work.
- Data feeds, alerting infrastructure, and observability tooling.

The cost is **fixed per investor**, irrespective of trading PnL. There is no performance fee, no carry, no hurdle. The alignment of interest is structural: the operator's revenue is the operating-cost reimbursement, which is bounded; the investor keeps 100 % of net PnL after costs.

Currency note. Trading capital is held in USDT on Binance. CHF amounts shown above are reference values for Swiss-resident investors; the actual auto-pay transfer is denominated in USDT against the prevailing CHF/USDT spot rate at the moment of transfer.

0.15.4 14.4 Performance projections

We project performance under three regime scenarios, on the basis of MVP data:

Market regime	Daily ROI projection	Annualised (linear, 365 days)
Drawdown	0.35 %	+127 % p.a.
Sideways	0.50 %	+180 % p.a.
Bullish	0.65 %	+237 % p.a.

These are **projections on the basis of MVP data**, not historical track records. (See Section 9.5 for the operational differentiators against the academic 15–30 % envelope.)

0.15.5 14.5 Capital-deployment scenarios

Three deployment scenarios are presented for due-diligence framing. Each scenario is evaluated under the three regime projections from Section 14.4. The CHF 100 k investment per slot is the **investor's contribution**; the **trading capital** column reflects scenarios in which investors choose to allocate additional capital beyond the contracted CHF 100 k slot.

Scenario 1 — Base case: CHF 100 k investment, CHF 100 k trading capital.

The investor's CHF 100 k contribution is the trading capital; no additional own-capital is deployed.

	Y1 PnL (after costs)	Y2 cumulative PnL (after costs)
Drawdown 0.35 %	−CHF 49 k	+CHF 2 k
Sideways 0.50 %	+CHF 5 k	+CHF 110 k
Bullish 0.65 %	+CHF 59 k	+CHF 218 k

Critical observation: Scenario 1 + Drawdown is the worst-case argument. In the most pessimistic regime scenario, year 1 ends at −CHF 49 k. By year 2 cumulative, the position has reached breakeven (+CHF 2 k). **Even in the worst-case projection, the investor is approximately whole at the 24-month mark.** This is the most important figure in the table for risk-averse due-diligence: the downside path on a stand-alone CHF 100 k commitment is approximately recoverable on a two-year horizon under the projected regime mix.

Scenario 2 — Doubled trading capital: CHF 200 k trading capital, CHF 100 k investment.

The investor allocates an additional CHF 100 k of own-capital alongside the slot, doubling the trading base while keeping the operating-cost share unchanged at CHF 75 k/year.

	Y1 PnL (after costs)	Y2 cumulative PnL (after costs)
Drawdown 0.35 %	+CHF 77 k	+CHF 254 k
Sideways 0.50 %	+CHF 185 k	+CHF 470 k
Bullish 0.65 %	+CHF 293 k	+CHF 686 k

The leverage of additional own-capital against the fixed cost share is the structural advantage of Scenario 2: the operating cost is the same, but the trading base from which percentage returns are generated is doubled.

Scenario 3 — High trading capital: CHF 500 k trading capital, CHF 100 k investment.

	Y1 PnL (after costs)	Y2 cumulative PnL (after costs)
Drawdown 0.35 %	+CHF 455 k	+CHF 1.01 M
Sideways 0.50 %	+CHF 725 k	+CHF 1.55 M
Bullish 0.65 %	+CHF 995 k	+CHF 2.09 M

These figures are pre-tax. They do not constitute a forecast of any individual investor's outcome; they are projections under the MVP-derived regime model. Scenario 3 is illustrative of the operating-cost-amortisation advantage at scale: the per-investor cost remains CHF 6,250/month, but the trading base it is set against is five times the contractual slot size.

All figures in CHF are reference values for Swiss-resident investors; actual position sizes, PnL, and operating-cost transfers occur in USDT against the prevailing CHF/USDT spot rate. Investors may deposit any amount they choose into their Binance account; the CHF 100 k figures shown are example deployments, not contractual minimum balances.

0.15.6 14.6 Onboarding flow

The onboarding sequence reflects the non-custodial architecture: KonnectAI never receives, holds, or moves investor funds. The investor controls the account, the deposit route, and the API key. KonnectAI controls only the trading logic that operates *through* the API key.

1. **Contract signature, Tranche 1 paid.** The investor signs the platform-access contract and pays Tranche 1 (CHF 50 k) by Swiss bank transfer or wire to the operator's banking entity. This is the platform-access fee structure, separate from trading capital.
2. **Investor opens / uses their own Binance account.** A standard retail Binance account is sufficient. Binance handles KYC and AML directly with the investor; KonnectAI is not a party to that process.
3. **Investor deposits CHF/USD via a Swiss fiat on-ramp** (e.g. SEBA, Sygnum, Bitstamp, Kraken Pro, or any regulated route they prefer) and converts to USDT inside their Binance account. The amount is entirely at the investor's discretion; CHF 100 k is the example baseline used in the projections.
4. **Investor creates an API key with the scoped permission profile** documented in Section 14.7: spot trading, futures trading, `Universal Transfer` / Binance Pay, **withdrawal disabled**, IP-whitelisted to the KonnectAI execution server.
5. **Investor delivers the API key and secret to KonnectAI through an encrypted channel** (PGP, Signal, Wire, or equivalent). KonnectAI ingests the key into the encrypted vault.
6. **KonnectAI registers the slot and runs a dry-run validation cycle.** A signal-echo run confirms the key is correctly scoped and IP-whitelisted, without placing live orders.
7. **Live trading commences within 24 hours** of vault ingestion and dry-run validation.

8. **One month after stable production trading, Tranche 2 is due** (CHF 50 k), settling the contractual platform-access total.

At every step, the investor controls the account, the deposit route, and the API key. The investor can revoke the API key in seconds via the Binance UI; revocation immediately stops all KonnectAI trading on that account.

0.15.7 14.7 Investor API-key permission profile

The API key the investor issues to KonnectAI has the following permission set, configured directly in the investor's Binance account:

```
Spot & Margin Trading:    ENABLED
Futures Trading:          ENABLED  (USDS-M)
Universal Transfer:         ENABLED  (for monthly operating-cost auto-deduct via Binance Pay)
Withdrawal:              DISABLED (KonnectAI never has withdraw rights)
IP Restrictions:          [KonnectAI server IP]
```

This profile is the contractual interface between investor and platform. It is **the** custody guarantee: even with full control of the API key, KonnectAI cannot move funds off-exchange. The worst-case impact of any compromise on the KonnectAI side is bounded by the no-withdrawal scope: a malicious actor in possession of the key could only execute trades on the investor's account, on which the investor can revoke the key in seconds.

0.15.8 14.8 Risk disclosure (abbreviated; full disclosure in the investor-subscription package)

- **Drawdown risk.** Even a well-tuned system will experience drawdown periods. The Section 14.5 Scenario 1 + Drawdown column is the projected worst case; actual drawdown periods may be deeper or longer than projected.
- **Regime-shift risk.** A regime transition the classifier does not capture can produce an outsized losing cluster. This is why the classifier is monitored as a first-order diagnostic (Section 11.2).
- **Operational risk.** The four-investor cluster shares a single operating environment. Operational incidents (LLM provider outage, exchange API outage, operator error) affect all investor accounts simultaneously. Mitigations are described in Section 12.4.
- **Liquidity risk.** At very large account equities, slippage on thinly traded altcoins becomes a materially larger drag, reducing effective returns. Watchlist composition and sizing are conditional on cluster-level capital scale.
- **Regulatory risk.** EU MiCA, Swiss FINMA, and US-equivalent regulatory developments may affect what venues and instruments are accessible, and may affect the legal positioning of non-custodial execution services. The current legal positioning is non-fund, non-custodial, non-managed-account; future regulatory developments may require

restructuring.

- **Provider risk.** Dependency on Anthropic (LLM), Binance (exchange), and on-chain data providers. Mitigation: multi-provider abstraction layers in the architecture, pre-built switchover paths.
- **Cluster-level concentration.** All four investors in a cluster share the same operating logic. Diversification across cluster operators is not in scope of this document; the second-cluster scale-out (Section 11.4) is the architectural answer to demand-side diversification.

0.15.9 14.9 Why this beats passive holding

In principle, buy-and-hold BTC over five-year windows has produced strong absolute returns — albeit with drawdowns in excess of −70 % in the 2022 cycle. KonnectAI Trader's rationale versus passive holding rests on three pillars:

1. **Continuity of performance.** The bidirectional execution layer profits in bull, sideways, and drawdown regimes. A passive position is exposed to the directional bias of the market over the holding period.
2. **Drawdown management.** Regime-aware execution with regime-change-exit is designed to limit drawdown depth. Passive holding has no such mechanism.
3. **Risk-managed, not directional.** The system is positioned as a risk-managed continuous-performance instrument. Passive holding is, by construction, a directional bet.

For investors whose utility function weights drawdown heavily, or who want exposure to crypto without the regime-blind directional risk, the active capital-access model is structurally different from a passive position.

0.16 15. MVP Status and Track Record

0.16.1 15.1 Operational since January 2026

KonnectAI Trader has run live, against Kai Zeh's own capital, on Binance Spot and Binance Futures, since **January 2026**. The current feature set has been in continuous production since **April 2026**. This section summarises what is operational and verifiable today, in advance of the four-investor production cohort.

0.16.2 15.2 What is in production today

Trading layer.

- Live trading on Binance Spot (long-only, in selected USDT pairs).

- Live trading on Binance Futures (long and short), with leverage and margin tracked per trade.
- Bidirectional execution: the analyser emits BUY (long) or SELL (short) actions; the execution layer routes accordingly.
- Partial-take-profit-with-runner across both Spot and Futures.
- Original-SL preservation across all trades; TP1 trail does not overwrite the original stop in the audit record.
- TP1-aware position sizing: risk per trade computed against entry-to-original-SL distance.
- Regime-change exit: auto-close on adverse regime flip when hypothetical PnL exceeds round-trip-fee threshold.
- Cooldown enforcement after stop-out, per symbol.

Accounting layer.

- Wallet-based net-PnL %, computed via geometric period-linking across capital-event boundaries.
- Capital-events sync with Binance: deposits, withdrawals, and Spot↔Futures internal transfers all reconciled into the equity curve, with idempotent re-sync via UNIQUE `external_id`.
- Internal-transfer bridge logic: phantom-dip elimination on the aggregate Spot/Futures equity curve.
- Asset-breakdown stacked-area chart on the dashboard portfolio view, with capital-event markers.
- Dust-threshold filtering on the portfolio view.
- Reconciliation-drift monitoring: continuous comparison of computed equity against Binance wallet snapshot.

Audit layer.

- Four primary trade-and-event repos: `live_trades`, `futures_trades`, `capital_events`, `internal_transfers`. Each row carries timestamp, payload, attribution, and pipeline version.
- Full LLM analysis archive with reasoning text, regime label, on-chain context, and strategy-mode tag.
- Migrations are additive; no destructive schema changes have been required.

Operations layer.

- 15-minute core-loop cron, running every cycle without missed executions for sustained windows.
- Hourly on-chain, macro, and portfolio snapshots.
- Twice-daily Telegram briefings.
- Weekly LLM-generated macro report.
- Live-KPIs dashboard reflecting current open trades, recent fills, equity-curve health.

- Health-check probes verifying scheduled-job completion and alerting on anomalies.

Dashboard.

- TOTP-authenticated private surface and sanitised public surface.
- Authenticated views: full analyses, full trades (live and futures), portfolio with bridge-fixed equity curve and capital-event markers, capital-events log, settings, macro reports archive.
- Public views: watchlist, charts, sanitised analysis distribution, on-chain summary, daily cost telemetry.

0.16.3 15.3 Position of strength

The distinction between this version of the whitepaper and earlier framings is structural. We are not seeking funding for an 18-month build-out; the build-out is complete. We are not asking investors to validate a thesis on academic literature; the thesis is being measured live, on real capital, today. The continuous-improvement and production-migration plan described in Section 13 is incremental work against an operational baseline, not a research project.

This is the position from which we offer the four-slot capital-access model: a working system, capacity-bounded by the four-investor fan-out economics, with a clean cost-and-revenue separation between investor PnL (which stays in the investor's own account) and operator stipend (auto-paid monthly via Binance Pay under investor-controlled toggle).

0.17 16. References

Public data providers referenced. Binance public market-data API; Binance Spot, Futures, and Pay APIs (`/sapi/v1/pay/transactions` for the auto-pay flow); Coin Metrics Community API; Etherscan V2 public API; blockchain.com charts API; mempool.space API; Bitnodes API; DefiLlama API; Alternative.me Fear & Greed API; CoinGecko public API.

LLM provider referenced. Anthropic Claude (Opus 4-7 family).

Academic frameworks referenced. Hidden Markov Models (regime classification literature); Reinforcement Learning (FinRL family); Reflective LLM-agent pattern (CryptoTrade).

0.18 Appendix A: Reference Implementation — The Operating System

The preceding sections describe KonnectAI Trader as a conceptual framework: what it is, why each component exists, how they compose. This appendix documents what we have actually built, running live since January 2026, with the current feature set in continuous production since April 2026. The purpose is to de-risk the investor decision: we are not

asking investors to fund a thesis on academic literature, we are offering capital-access slots on a working operating system whose architectural soundness can be verified directly.

0.18.1 A.1 Implementation overview

The live system consists of two coordinated services:

- **An analysis and trading backend** (Python-based, package `crypto_agent`). Runs on a scheduled-job model. Owns data ingestion, indicator calculation, regime classification, screening, LLM-based analysis, live and futures trading, paper-trading sandbox, capital-events sync, regime-change-exit logic, alerting, briefings, and the weekly macro-report generator.
- **A web dashboard** (Node.js / Express / EJS). Serves the operator UX described in Section 10, with a TOTP-authenticated private surface and a sanitised public surface. Reads from the same persistence store as the backend.

Both services run on a single AWS EC2 Ubuntu instance at MVP scale, behind a Caddy reverse proxy with Let's Encrypt-managed TLS. Data persists in a PostgreSQL database on the same instance. Observability is via structured logs and journald.

The production deployment for the four-investor cohort separates compute tiers (analysis + execution + dashboard on dedicated production hardware), runs the core loop at 1-minute cadence (versus 15-minute on the MVP), and operates on **investor-owned Binance accounts** via per-investor scoped API keys held in an encrypted vault — KonnectAI never holds custody of investor funds (Sections 11.4, 12, 14).

0.18.2 A.2 Technology stack (concrete)

Backend services (Python). Modular package layout under `crypto_agent.src`:

- `market_data.py` — Binance candle ingest at multiple intervals; indicator computation (RSI, MACD, Bollinger, EMAs, ATR, volume ratios) on completed-candle baseline.
- `indicators.py` — indicator library, kept stateless and pure for testability.
- `prefilter.py` — deterministic rule-based screener; emits candidates per cycle.
- `screener.py` — multi-trigger composition logic.
- `regime.py` — 5-state Kalena classifier on the 4-hour candle stream; emits one row per `(symbol, captured_at)` per cycle.
- `regime_exit.py` — regime-change-exit helper (described in A.4).
- `analyzer.py` — LLM analyser; constructs the structured prompt and dispatches to Anthropic.
- `anthropic_client.py` — LLM client wrapper with retry and schema validation.
- `onchain.py` — multi-source on-chain snapshot sync.
- `macro.py` — macro and sentiment snapshot sync.
- `paper_trading.py` — MVP-only paper-trader sandbox.

- `live_trading.py` — Spot live-trading: place / track / reconcile / close.
- `futures_trading.py` — Futures live-trading (long and short): leverage, margin, funding.
- `capital_events.py` — Binance deposit / withdraw / internal-transfer sync (described in A.3).
- `binance_client.py` — Binance API client with retry, rate-limit, and per-investor-key scoping (one client instance per investor key, looked up from the vault).
- `portfolio.py` — portfolio aggregation, dust-threshold filtering, asset-breakdown view.
- `briefings.py`, `alerts.py`, `telegram.py` — reporting layer.
- `cli.py` — operator CLI for ad-hoc invocations.
- `db.py` — PostgreSQL connection pool + helpers.

Database. PostgreSQL with relational schema for price and indicator data, JSONB columns for richer payloads (analysis reasoning, regime metrics, on-chain raw snapshots, trade notes). Indexes are keyed on symbol+timestamp for ingest hot-paths, and on analysis/trade id for the dashboard read-paths.

Migrations. Append-only additive migrations checked into `agent/scripts/`. Selected migrations: - `migrate_v2.sql`, `migrate_phase2.sql` — baseline schema evolution. - `migrate_regime.sql`, `migrate_regime_exit.sql` — regime persistence + exit branch. - `migrate_live_trading.sql`, `migrate_futures_trading.sql` — live trade tables. - `migrate_capital_events.sql` — capital-events + internal-transfers tables. - `migrate_partial_tp.sql`, `migrate_original_sl.sql` — TP/SL handling refinements. - `migrate_emergency_close.sql`, `migrate_emergency_close_futures.sql` — manual-close audit. - `migrate_dust_sweep.sql` — portfolio dust-threshold support. - `migrate_macro.sql`, `migrate_onchain.sql`, `migrate_prefilter.sql` — supporting tables. - `backfill_original_sl.sql` — historical backfill of `original_SL` from pre-preservation trades.

LLM. Anthropic Claude Opus 4-7 via the standard Anthropic API. Structured-output prompts with a JSON schema; single-retry policy on schema-invalid responses (rarely exercised in production).

Scheduling. Combination of systemd timer units (long-running services) and cron (the core loop and scheduled briefings). MVP runs on a 15-minute core loop. Production migrates to a 1-minute core loop.

Frontend. Node.js / Express / EJS server-rendered HTML. TradingView `lightweight-charts` for candle/indicator rendering; Chart.js for equity curves and distribution charts. Server-rendered with targeted client-side enhancement — a deliberate choice for simplicity and operator-friendly performance. Cache headers explicitly disabled on HTML and API responses to avoid stale-data confusion in the operator's browser session.

Deployment. AWS EC2 Ubuntu LTS. Caddy reverse proxy. Let's Encrypt TLS. Application-

layer secrets in a credentials directory with OS-level permission restrictions.

Observability. Structured logs via journald sink, plus an application-level log file. Periodic health-check scripts verify scheduled-job completion and alert via Telegram on anomalies.

0.18.3 A.3 Capital-events pipeline

The `capital_events.py` module powers wallet-based net-PnL via geometric period-linking. Three Binance sources are pulled and reconciled:

- `get_deposit_history` (status=1 → success).
- `get_withdraw_history` (status=6 → completed).
- `get_universal_transfer_history` (Spot↔Futures, types `MAIN_UMFUTURE` and `UMFUTURE_MAIN`) — stored separately as audit-only.

Critical Kai-directive: internal Spot↔Futures transfers are **never** counted as capital events. They land in the `internal_transfers` table only. Treating internal transfers as deposits or withdrawals would corrupt the equity-curve methodology.

USD-equivalent pricing. For non-USDT capital events, the USD equivalent is fetched via 1-minute kline close on `{asset}USDT` at the precise `occurred_at` timestamp. This produces historically accurate USD values rather than current-price approximations. Stablecoins (`USDT`, `BUSD`, `USDC`, `FDUSD`, `TUSD`, `DAI`) are treated as USD-pegged at $1.

Idempotency. Every row carries a Binance-side `external_id` marked UNIQUE. The sync is safe to re-run as often as desired without duplicating capital events. Schematically:

```
# capital_events.py (excerpt, simplified)

USDT_ASSETS = {"USDT", "BUSD", "USDC", "FDUSD", "TUSD", "DAI"}
DEPOSIT_STATUS_SUCCESS = 1
WITHDRAW_STATUS_COMPLETED = 6
TRANSFER_SPOT_TO_FUTURES = "MAIN_UMFUTURE"
TRANSFER_FUTURES_TO_SPOT = "UMFUTURE_MAIN"

def _get_usd_price_at(asset: str, ts_ms: int) -> float | None:
    """Historical USD price via 1m kline close at ts_ms."""
    if asset in USDT_ASSETS:
        return 1.0
    # ... 1m kline lookup against {asset}USDT, returning close at ts_ms

def sync_capital_events() -> None:
    # 1) Pull deposits, withdrawals via Binance Spot endpoints.
    # 2) Compute usd_value at occurred_at via _get_usd_price_at.
    # 3) Upsert into capital_events (external_id UNIQUE).
```

```
    # 4) Pull universal-transfer history; store in internal_transfers only.
```

Wallet-based return computation. With capital events partitioning the equity curve into deposit-aware periods, the per-period return is:

`r_i = (equity_end_i − equity_start_i − net_capital_event_i) / equity_start_i`

and the total return is the geometric chain:

`total_return = (1 + r_1) × (1 + r_2) × ... × (1 + r_n) − 1`

This is the figure reported on the dashboard equity-curve view as **wallet-based net-PnL %**. It survives deposits and withdrawals, which is the property that distinguishes a defensible return claim from a naive equity-delta claim.

0.18.4 A.4 Regime-change-exit helper

The `regime_exit.py` module is the executable form of the regime-change-exit branch (Section 5.5). Pure-function design: it computes whether a close is required; the caller (`live_trading.update_live_trades` and `futures_trading.update_futures_trades`) performs the actual close.

Key constants and mappings:

```
# regime_exit.py (excerpt)

# Single source of truth for taker fees.
SPOT_TAKER_FEE_PCT    = 0.075   # 0.10% standard, 0.075% with BNB-discount
FUTURES_TAKER_FEE_PCT = 0.04    # 0.05% standard, 0.04% with BNB-discount

# Internal Kalena labels → coarse direction.
_REGIME_DIRECTION = {
    "trend_up_liquid":   "bull",
    "trend_down_liquid": "bear",
    "crash":             "bear",       # treated as strong against-LONG
    "range":             "sideways",
    "vol_expansion":     "sideways",   # ambiguous → safer: no action
}

NEUTRAL_DIRECTIONS = {"sideways", "neutral", "unknown", None}
```

The exit decision logic, in plain terms:

1. Map the current regime to a direction (`bull`, `bear`, `sideways`, `unknown`).
2. If neutral or unknown: skip (no action).
3. If the direction now opposes the trade side (long-vs-bear, or short-vs-bull):
    - Compute hypothetical realised PnL % at current price, on **notional** (`qty` ×

price).
- Compare against round-trip taker fees (entry + exit).
- If hypothetical PnL exceeds the round-trip-fee threshold: close the trade with exit_reason='regime_change'.
- If hypothetical PnL is below threshold: leave the original stop and TP active.

Leverage note. Fee/funding math is on notional, which is leverage-agnostic. The futures leverage setting does not enter these calculations — verified explicitly during the live-trading rollout.

Asymmetric-loss property. The branch never fires on a losing trade. It cannot turn a stopped-out trade into an earlier loss. Its only function is to lock in profit when a profit-bearing position is exposed to a hostile freshly classified regime where continuation expectation has flipped negative.

0.18.5 A.5 Equity-curve bridge for internal transfers

The production deployment surfaces a unified Spot+Futures equity curve to the operator. Internal Spot↔Futures transfers are asynchronous events that, naively summed across snapshot windows, produce **phantom dips** — momentary apparent equity drops as the transferred amount briefly disappears from one wallet's snapshot before appearing in the other's.

The bridge logic, applied at chart-rendering time on the dashboard:

1. Pull all internal_transfers rows with their (occurred_at, from_wallet, to_wallet, asset, amount, usd_value).
2. For each adjacent equity-snapshot pair, classify whether an internal transfer falls between them.
3. If yes, attribute the transferred amount to the source wallet at the to-time, eliminating the phantom dip on the aggregate curve.
4. The two component curves (Spot only, Futures only) remain unbridged — each shows its own real movement; the bridge applies only to the aggregate.

The practical effect is a smooth aggregate curve through the moment of internal-transfer execution, with capital-event markers (real deposits and withdrawals) overlaid as discrete points.

0.18.6 A.6 Original-SL preservation and TP1-aware sizing

Two refinements applied to both Spot and Futures live-trading:

Original-SL preservation. When TP1 is reached, the runner stop is trailed up to the TP1 level (locking in the TP1 gain on the runner leg). However, the *original* stop level is preserved as original_SL on the trade record. Two reasons:

1. **Risk attribution.** Position sizing was computed against the entry-to-original-SL distance. Post-hoc analysis of "what was the maximum adverse excursion against the original-risk premise" requires the original stop to be queryable.
2. **Audit integrity.** A reader of the trade record after closure needs to be able to reconstruct the trade as the strategy actually placed it, not as it evolved.

The `migrate_original_sl.sql` migration adds the `original_SL` column; `backfill_original_sl.sql` populates it for historical trades.

TP1-aware position sizing. Risk per trade is computed as `(entry − original_SL) × qty`, not as `(entry − stop_active) × qty`. This matters because once TP1 has trailed the active stop up, the active-stop-distance becomes the post-trail distance. Sizing on that post-trail distance would systematically over-size new trades. Sizing on the original-SL distance keeps the risk envelope consistent with the strategy's pre-trade intent.

0.18.7 A.7 Pooled four-investor non-custodial fan-out architecture

The production deployment for the four-investor cohort runs on **investor-owned Binance accounts** — not on Binance sub-accounts under a KonnectAI master. The sub-account route would require Binance VIP status (~$3.8 M+ holdings or $50 M+ monthly volume), which we do not have, and would put investor funds under our umbrella, which is exactly the custodial posture we refuse to take. The design instead is non-custodial by construction: each investor holds and controls their own standard-retail Binance account; KonnectAI operates *through* a scoped API key the investor issues.

Setup (per investor).

1. The investor opens (or uses an existing) standard-retail Binance account. KYC and AML are handled directly between the investor and Binance.
2. The investor deposits CHF/USD via a Swiss fiat on-ramp (e.g. SEBA, Sygnum, Bitstamp, Kraken Pro) and converts to USDT inside their account. The amount is at the investor's discretion; CHF 100 k-equivalent USDT is the example baseline used in the projections.
3. The investor creates a Binance API key with the contractual permission profile: spot & margin trading, USDS-M futures trading, `Universal Transfer` / Binance Pay (for the monthly operating-cost auto-deduct), **withdrawal permanently disabled**, IP-whitelisted to the KonnectAI execution server.
4. The investor delivers the API key + secret to KonnectAI through an encrypted channel (PGP, Signal, Wire). KonnectAI ingests it into the encrypted API-key vault.
5. The investor's user record in the auth store is bound to the corresponding vault slot; their dashboard view filters all data by their vault-slot identifier.
6. KonnectAI runs a dry-run validation cycle (signal echo, no live orders) to verify the key is correctly scoped and IP-whitelisted, and live trading commences within 24 hours.

Operating loop (per cycle).

```
for each cycle:
    candidates = screener.scan(watchlist)
    for candidate in candidates:
        analysis = analyzer.analyse(candidate)        # 1 LLM call
        decision = validators.evaluate(analysis)
        if decision.action in (BUY, SELL):
            for investor_account in subscribed_investor_accounts:
          # investor_account holds (vault_slot_id, api_key_handle, equity, policy)
                size = sizer.size_for(
             investor_account.equity, decision, investor_account.policy)
                executor.place_order(investor_account, decision, size)
        persist(analysis, decision, fills)
```

The LLM call is **once per candidate per cycle**, not per investor account. The fan-out is at the execution layer, via four parallel `binance_client` instances each bound to a different investor's API key, where size scales with the equity available on that investor's account. Per-investor LLM cost = TotalLLM / 4.

Operating-cost auto-pay (Binance Pay). Once per month — on the 1st at 09:00 Europe/Zurich — the operating-cost amount (CHF 6,250-equivalent USDT) is moved from each investor's account to the dedicated KonnectAI operator wallet via the Binance Pay endpoint (`/sapi/v1/pay/transactions`). The transfer is idempotent (UNIQUE `external_id`) and audit-logged identically to capital events. The flow is gated by an **investor-controlled settings toggle** (default ON): if the investor turns it OFF, the system pauses trading on that account until the operating cost has been settled manually. Pay-as-you-go: no pre-payment, no refund, no carry.

Profit distribution. By default, net-PnL stays in the investor's account every cycle (compound by default). The investor may, in settings, enable an optional monthly auto-distribution of either a fixed percentage or a fixed amount of net-PnL to an external cold wallet of their choice. There is no fund-style "distribution" because there is no fund; what is in the investor's account *is* their balance.

Custody guarantee. No KonnectAI-held credential, on any investor account, can move funds off-exchange. This is enforced at the Binance permission level (Withdrawal: DISABLED), not in our application logic. The investor can revoke the API key at any moment via the Binance UI; revocation is instantaneous and immediately stops all KonnectAI trading on that account.

Hard cap. The architecture enforces a maximum of four investor accounts per cluster. Beyond four, fan-out variance and round-lot inefficiencies degrade execution quality. The scaling answer is to deploy a second cluster (separate hardware, separate operator wallet, separate four-investor pool of investor-owned accounts) rather than to grow a single cluster vertically.

0.18.8 A.8 Cron-cadence migration: MVP 15-min → Production 1-min

The MVP runs on a 15-minute core-loop cron (96 cycles/day). The production deployment runs on a 1-minute core-loop cron (1,440 cycles/day). The migration is operational, not architectural — the same code paths, the same database schema, the same execution logic. What changes:

- **Candle ingest cadence.** The 1-minute candle stream is added to the ingest set on production. The MVP synchronises to 15-minute candle close; production synchronises to 1-minute candle close.
- **LLM-inference budget.** The production cluster is provisioned with an inference-cost budget consistent with 1,440 cycles/day across the four-investor-account fan-out. Budget alerts are calibrated accordingly.
- **Indicator-calculation hot path.** The 1-minute timeframe joins the existing 15m / 1h / 4h / 1d set on the production deployment.
- **Health-check cadence.** Health-check probes run more frequently in production to keep mean-time-to-detection bounded against the higher cycle rate.

The higher cron granularity is one of the structural reasons our projected returns sit above the academic 15–30 % p.a. envelope (Section 1.3, 9.5): the inference frequency is two orders of magnitude higher than published designs.

0.18.9 A.9 Architectural decision records (selected)

The following ADRs are documented in the project's engineering notebook. The most investor-relevant:

ADR — Opus-class LLM for the analyser. We chose Anthropic Claude Opus 4-7 over cheaper Sonnet-class alternatives, after internal comparison runs on historical candidate sets. The cost delta (approximately 4–5× per call) is offset by materially better multi-signal reasoning quality in our observed outputs. The framework is model-agnostic and substitution is a prompt-and-schema adaptation exercise.

ADR — Partial-TP-with-runner. We adopted partial-TP-with-runner (Section 8.2) in preference to either full-exit-at-TP1 or single-target-TP2. Driven by observed behaviour in early live trading: TP2 is rarely reached without TP1 being reached first, and full-exit-at-TP1 forgoes upside in trending regimes.

ADR — Strategy mode as a global database setting. Operator-switchable without code change, persists across service restarts, supports strategy-mode tagging on trades for A/B analysis. Per-symbol modes were rejected as prematurely complex; parallel-shadow strategies were rejected on cost (multiplied LLM inference).

ADR — TOTP-only authentication. Passwords are the dominant breach vector in small-team web applications; eliminating them removes that surface. TOTP secret is generated once per user, displayed as QR for authenticator-app enrolment, stored server-side.

ADR — Internal Spot↔Futures transfers are NEVER capital events. Counting them would corrupt the equity-curve methodology by double-counting capital that has merely changed wallets. They are stored in `internal_transfers` for audit, never in `capital_events`. (See A.3.)

ADR — Original-SL preservation under TP1 trail. TP1 trail updates the active stop for execution purposes, but the original stop is preserved on the trade record for risk-attribution and audit purposes. (See A.6.)

ADR — Regime-exit math on notional, not margin. The fee-and-funding threshold for regime-change exit is computed on notional (`qty × price`), which is leverage-agnostic. Dividing by margin would produce different exit triggers at different leverage levels for economically equivalent positions, which is wrong.

ADR — Coin Metrics Community API as the early-phase exchange-flow source. Glassnode's free tier does not expose a programmatic API; Coin Metrics Community is keyless and provides core exchange-inflow/outflow metrics. Migration path to Glassnode Pro is part of the post-MVP enhancement roadmap.

0.18.10 A.10 Current production status

As of May 2026, in production:

- **Live trading on Binance Spot (long-only).** Operational since January 2026. Real fees, real slippage, real PnL.
- **Live trading on Binance Futures (long and short).** Operational. Bidirectional execution validated.
- **Six on-chain sources** snapshotted hourly: blockchain.com, mempool.space, bitnodes, DefiLlama, Coin Metrics Community, Etherscan.
- **5-state regime classifier** running on the 4-hour candle stream, persisted per cycle.
- **Regime-change exit** branch wired into both Spot and Futures execution paths; logs `exit_reason='regime_change'` on trigger.
- **Capital-events sync** running on every cycle; deposits, withdrawals, and Spot↔Futures internal transfers reconciled with idempotent `external_id`.
- **Wallet-based net-PnL %** displayed on the dashboard equity-curve view, computed via geometric period-linking.
- **Internal-transfer bridge** applied at chart-render time; phantom dips eliminated on the aggregate equity curve.
- **Asset-breakdown stacked-area chart** with capital-event markers visible on the portfolio view.
- **Dust-threshold filtering** at $5 on the portfolio breakdown.
- **Original-SL preservation** active on all new trades; historical trades backfilled.
- **TP1-aware position sizing** applied across Spot and Futures.
- **Dashboard** live with TOTP authentication; private and public surfaces both functional.

- **Telegram briefings** twice daily, weekly macro report on schedule.
- **Health-check probes** verifying scheduled-job completion; alert-on-anomaly via Telegram.
- **Cost envelope** consistent with the operating-cost target on the MVP cron; production-cron envelope is provisioned for the four-investor cohort's 1-minute cadence.

0.18.11 A.11 Known limitations and ongoing work

Investor-facing honesty requires explicit acknowledgement of limitations:

- **Ethereum smart-contract-custody under-tracking.** Coinbase and Kraken use smart-contract custody architectures whose addresses are not consistently labelled in the free-tier Etherscan label set. Our aggregate tracked-ETH figure for these exchanges is a fraction of the true custody level. *Relative* movements remain informative; *absolute* levels should be treated as lower bounds. Resolution path: Arkham Intelligence label-database integration (paid, post-MVP enhancement).
- **Slippage model in paper trader.** The MVP paper trader does not simulate slippage; it remains a research bench for prompt and threshold revisions, not a performance validator. The live system measures slippage trade-by-trade; live numbers are the only performance numbers we report against.
- **Regime-conditional execution refinements.** The regime classification is consumed in the analyser prompt and in the regime-change-exit branch. Additional regime-conditional refinements (regime-conditional sizing, regime-conditional TP ladders) are on the continuous-improvement track.
- **Social-sentiment integration.** Not present. Future post-MVP enhancement; requires semantic-deduplication discipline to avoid echo-chamber over-weighting.
- **Active-addresses 7-day trend.** Requires accumulated historical snapshots before it is usable as an analyser-prompt input. The snapshot-accumulation path is in flight.
- **Single-region deployment.** MVP runs in one AWS region. Production deployment includes multi-region consideration, particularly for the ingest tier.
- **Telegram-media deduplication patch.** A small workaround is maintained at the OpenClaw extension layer for a duplicate-send behaviour. Marker-and-verification-script discipline ensures it can be removed at upstream resolution.

0.18.12 A.12 Evidence of execution

The live system is accessible for investor demonstration at an authenticated domain. The public surface is viewable without credentials. The private surface — including full trade detail, equity curves with capital-event markers, the LLM reasoning archive, and the regime-classification persistence — is available under NDA to qualified investor representatives.

What the demonstration shows:

- A system that runs continuously without operator intervention.

- Live Spot and Futures trades opening, scaling to TP1, and closing at stop / TP1 / TP2 / regime_change / timeout / manual across the expected exit reasons.
- Regime classification persisted at MVP cadence, visible per symbol and per cycle.
- LLM analyses generated on each candidate, with reasoning text that references regime, on-chain, and technical context.
- Capital events (deposits, withdrawals) and internal transfers reconciled into the equity-curve view, with the bridge logic eliminating phantom dips.
- Wallet-based net-PnL % displayed prominently on the equity-curve view, computed via geometric period-linking.
- Dashboard operator flows (settings change, trade close, capital-events log, macro-report browse) functioning end-to-end.
- Telegram briefings arriving on schedule.
- A daily cost envelope visible in public telemetry, meeting the stated MVP target.

This is not a slideware prototype. It is a complete, running operating system on which Kai Zeh has been trading his own capital since January 2026. The four-investor production cohort is an onboarding-and-scale-out exercise against this baseline, not a build-out.
