PRE-BENCHMARK RESEARCH PROPOSAL

NO VALIDATED FORECASTING OR PROFIT CLAIM

TECHNICAL WHITEPAPER · VERSION 0.2

HEATWAKE

The Liquidity Wavefunction

A dual-view latent model of order-flow markets with a separately tested observability layer.

By **William Keenan**
K&E Studios Research

July 10, 2026

13,135 words · 53 min

63 referenced sources

CONCEPTUAL ARCHITECTURE · NOT A MODEL OUTPUT

The market remains data until the last mile.

LWM-1 / SYSTEM MAP

NATIVE VIEW

Received event tape

Adds, changes, deletes, trades, timing, and integrity state.

FIELD VIEW

Deterministic semantic field

The same causal history rendered across time × relative price channels.

FUSE

PREDICTIVE STATE

Latent liquidity distribution

A recurrent state that preserves several conditional futures without turning them into prose.

MEASURE

FORECAST

Measurement heads

Lifecycle hazards, paths, fills, and settlement probabilities.

OBSERVABILITY

Rosetta

Future fields, concepts, analogues, sensitivity, and constrained explanation.

DECISION BOUNDARY

Deterministic policy

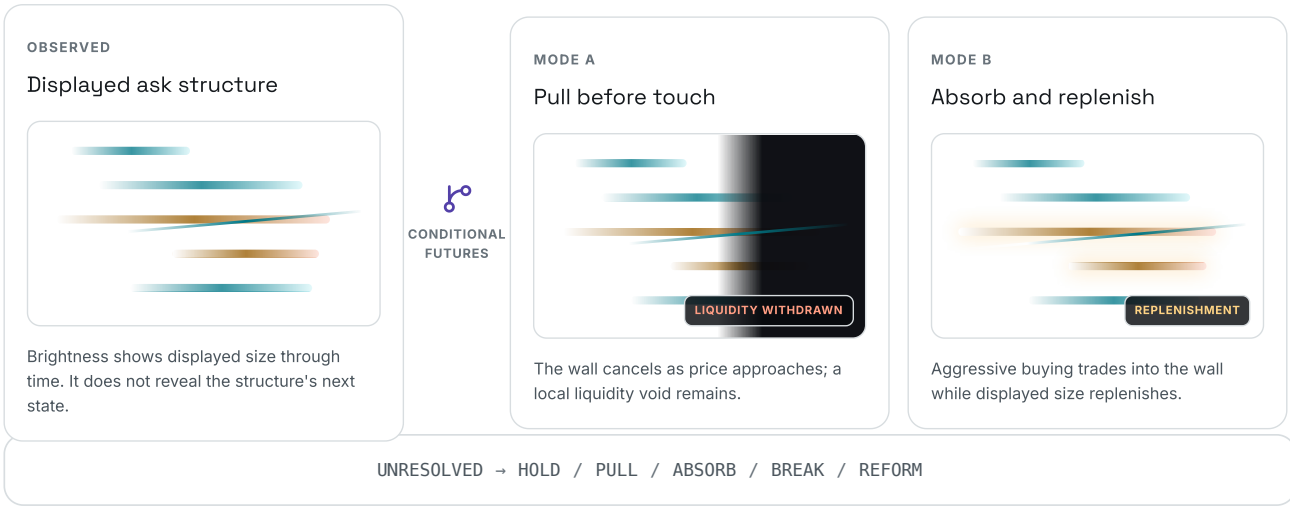
Health, calibration, executable cost, uncertainty, and risk gates.

The latent distribution is the predictive state. Rosetta and the interface are measured translations of it—not substitutes for it.

ILLUSTRATIVE MECHANISM · PROBABILITIES DELIBERATELY OMITTED

One visible wall. More than one possible movie.

MULTISTATE STRUCTURE



These are proposed state semantics, not observed HEATWAKE results. The experiment must establish whether they are predictable and useful beyond baselines.

Abstract

Market depth is visible. Its consequence is not.

Limit-order-book heatmaps turn additions, cancellations, executions, and resting liquidity into a spatiotemporal field. They expose the auction, but they do not determine whether a bright liquidity wall will hold, pull, replenish, absorb pressure, or fail. That interpretation remains a difficult human task, and the existence of repeatable predictive skill in it is an empirical question.

HEATWAKE proposes a domain-specific model that learns this field directly from native market messages. It does not take desktop screenshots every millisecond, reduce the market to hand-written indicators, or ask a language model to describe a chart. A deterministic renderer converts a checksum-validated, depth-bounded event log into synchronized semantic channels. A dual-view encoder reads both received market messages and the rendered liquidity field. A recurrent latent dynamics model then maintains a compact state over multiple plausible market futures.

We call the learned distribution over those futures the **Liquidity Wavefunction**. The term is explicitly quantum-inspired, not a claim that markets are quantum systems or that quantum hardware is required. In the production design, it is a classical conditional distribution over future liquidity trajectories. Incoming evidence may concentrate, disperse, or split its probability mass.

The model's native state is not English, a list of named scenarios, or a stream of rendered pixels. It is a continuous latent representation trained to retain the information needed to predict future order flow, liquidity-structure lifecycles, executable price paths, and—where applicable—binary market settlement. A separate observability system, called **Rosetta**, turns that latent state into future heatmaps, constrained concepts, sensitivity tests, nearest historical analogues, calibrated probabilities, and human-readable explanations. Human legibility is a separately tested output rather than a serialization constraint on the predictive state.

The project is falsifiable. HEATWAKE must beat event-only, field-only, engineered-feature, and established deep limit-order-book baselines on untouched chronological data. Any economic claim must additionally beat the prediction market's size-dependent arrival-quote baseline after fees, latency, fills, and model uncertainty. If the visual inductive bias or latent-future objective provides no incremental predictive value, or if apparent advantage disappears under execution, the relevant thesis is rejected.

Reader map

For the five-minute version, read the model in one page, the modeling problem, the Liquidity Wavefunction, Rosetta, the scientific contract, and the Bookmap proposition.

The complete paper also specifies data and platforms, training, failure criteria, the visual product, implementation and costs, the roadmap, and governance.

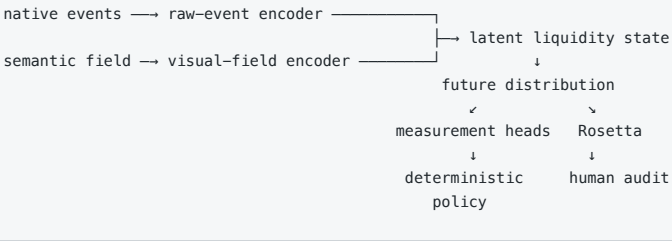
1. The model in one page

Rosetta may render an **IF/THEN graph**: a human-readable map of measured conditions, forecast changes, and invalidation rules. That graph is not the core model state.

The founding principle is:

> **The latent distribution—not its decoded explanation—is HEATWAKE's predictive state.**

The visual overlay is also a translation. Rosetta converts the latent distribution into inspectable future fields, measured concepts, historical analogues, sensitivity/intervention tests, and constrained prose.



1.1 A running example

At an observed ask wall, two coherent modes may dominate:

1. The wall pulls before price arrives, leaving a liquidity void and increasing continuation risk.
2. Price reaches the wall, aggressive buying trades into it, and replenishment produces absorption.

HEATWAKE does not average those modes into one blurry future. It maintains both with separate weights. If visible orders begin cancelling while cross-venue buying accelerates, the pull mode gains mass. If executions increase but visible size replenishes and price fails to cross, the absorption mode gains mass.

Rosetta then exposes the measured change:

```
PULL-BEFORE-TOUCH HAZARD      0.31 → 0.58
ABSORPTION HAZARD            0.44 → 0.23
SCENE FAMILIARITY            in distribution
DATA HEALTH                  valid
EXECUTABLE EDGE              below threshold
POLICY                       WAIT
INVALIDATION                 replenishment resumes at the tracked level
```

The example is illustrative, not a model result. Its purpose is to define the mechanism and the interface contract.

1.2 What changes relative to existing approaches

APPROACH	SOURCE FIDELITY	FUTURE REPRESENTATION	MULTIPLE FUTURES	HUMAN AUDIT	ACTION GATING
Engineered-feature pipeline	Native events collapsed into selected scalars	Direct label or score	Usually no	Feature-level	Varies
Screenshot/image forecaster	Display pixels	Class or generated pixels	Sometimes	Visually intuitive but artifact-prone	Often separate
Event-only temporal model	Native event sequence	Direct labels or generated events	Model-dependent	Difficult	Varies
HEATWAKE proposal	Native events plus deterministic semantic field	Latent trajectory distribution	Required	Rosetta + provenance + interventions	Calibration, quotes, costs, health, abstention

2. What HEATWAKE is

HEATWAKE names the proposed research and product system.

NAME	MEANING
HEATWAKE	The recorder, model, evaluation system, policy, observability layer, and interface.
LWM-1	The proposed first Latent Liquidity World Model checkpoint family. “World model” remains provisional until the model supports validated intervention-conditioned dynamics.
Liquidity Wavefunction	The model's evolving classical belief distribution over future liquidity states.
Latent liquidity language	A metaphor for the learned continuous state in which the model represents market structure before translating it for humans.
Rosetta	The separately tested decoder and audit surface.
Measurement heads	Calibrated forecasts derived from the core state: hazards, paths, fills, and settlement probabilities.
Policy layer	A separate deterministic gate that produces BUY_UP, BUY_DOWN, WAIT, or ABSTAIN from forecasts, quotes, costs, health, and risk rules.

Technically, the first model should be described conservatively as a **dual-view conditional liquidity-dynamics model**. A passive historical forecaster estimates:

$$p(o_{t+1:t+H} \mid o_{\le t}).$$

Here (o_t) denotes observable market events at time (t). A stronger world model would additionally learn how the market responds to an action:

$$p(s_{t+1}, o_{t+1} \mid s_t, a_t).$$

Here (s_t) denotes a learned market state and (a_t) a specified order or policy action.

Until intervention-conditioned dynamics are tested, LWM-1 is a product/model name rather than a claim that HEATWAKE has solved interactive market simulation.

The acronym **LCM** is deliberately avoided in the technical specification because “Large Concept Model” already names a published sentence-representation model family. **LWM-1** is more precise: the proposed system learns latent liquidity dynamics and conditional futures; it is neither a language model nor a candlestick model.

2.1 Current state

The project currently has:

- A simulated public interface.
- A v0.1 order-flow research proposal.
- The Liquidity Wavefunction design doctrine.
- This unified technical specification.

It does not yet have:

- Signed market-data rights.
- A depth-bounded, integrity-checked training corpus.
- A production recorder or shared tensor core.
- A trained LWM-1 checkpoint.
- A locked benchmark result.
- A validated user workflow or commercial claim.

2.2 First user and product wedge

The first intended user is an experienced discretionary order-flow trader who already understands liquidity heatmaps but must interpret them manually under time pressure. Market-microstructure researchers are validation collaborators, not the initial product persona. The first product is read-only decision support: a future-field and abstention overlay with replay, analogues, and audit—not an autonomous trading bot.

The initial job is:

> Convert a fast, discretionary reading of displayed liquidity into a calibrated, inspectable distribution over what the structure may do next.

Possible later products include a paid platform add-on, a research terminal, or a licensed model/API. Pricing, willingness to pay, and market size remain customer-discovery questions rather than whitepaper claims.

If a defensible moat emerges, it is expected to come from the synchronized licensed corpus, objective liquidity-lifecycle labels, live/replay parity, accumulated calibration history, and platform integrations—not from the heatmap renderer alone.

2.3 What it is not

HEATWAKE is not:

- An LLM prompted to analyze Bookmap screenshots.
- A large candlestick classifier.
- A collection of hand-written trading indicators.
- A claim that the market has a predetermined movie waiting to be discovered.
- A claim that bright liquidity is automatically support or resistance.
- A system for identifying individual traders from an aggregated heatmap.
- A claim that a generated future image is true because it looks plausible.
- A quantum computer, a quantum-speedup claim, or evidence that markets obey quantum mechanics.
- A live trading system before legal eligibility, data rights, locked evaluation, and shadow operation are satisfied.

2.4 The novelty claim we are willing to test

HEATWAKE is not the first image-based order-flow model, deep LOB model, generative LOB model, financial “world model,” or LOB foundation model. Prior work already covers each category 1–10.

The proposed contribution is an integration. Native events and a deterministic semantic field are encoded jointly. The model preserves a multimodal distribution over future liquidity states and estimates transition hazards for tracked structures. A separate policy requires executable expected value, uncertainty, healthy data, and risk clearance before action.

This paper makes no scientific priority claim. Its hypothesis is that the integration provides measurable forecasting, observability, and workflow value beyond its individual components.

CLOSEST PRIOR DIRECTION	WHAT IT DEMONSTRATES	GAP HEATWAKE PROPOSES TO TEST
DeepLOB and event/LOB forecasters	Spatial and temporal LOB structure can be predictive in reported datasets.	Dual event/field fusion, multimodal future fields, and live audit.
Order-flow image prediction	Bitcoin order flow can be encoded as image channels for volatility forecasting.	Structural lifecycle hazards and execution-aware directional evaluation.
Generative LOB and structured-image diffusion	Future order-book states can be generated and benchmarked.	A live perception state, calibrated measurement heads, and a human observability layer.
LOB world agents/foundation models	Market simulation and large pretrained LOB representations are active research areas.	Product-integrated conditional perception rather than a claim to invent the category.
Latent reasoning and visual world models	Models can predict or recur in continuous latent representations.	Whether that design survives adversarial, nonstationary market microstructure.
Concept/probe interpretability	Human concepts can sometimes be decoded and intervened on.	A separately scored Rosetta layer tied to market forecasts and invalidation.

2.5 Compact glossary

TERM	MEANING IN THIS PAPER
LOB	Limit order book.
CLOB	Central limit order book.
L2 / MBP	Aggregated quantity by price level; no individual visible-order identity.
L3 / MBO	Individual visible orders where the provider genuinely supplies them; depth and hidden-liquidity limits still apply.
Semantic field	A deterministic multichannel time × relative-price tensor rendered from causal feed messages.
Unresolved future field	A classical distribution retaining several weighted future modes rather than committing to one path.
Future-field haze	A display of decoded probability mass across future time and relative price; not a claim that every lit region will trade.
Concentration ridge	A connected region in that display where several weighted future trajectories place substantial mass.
Fragility zone	A region whose forecast changes materially under small, valid perturbations or nearby on-manifold latent interventions.
Availability time	The local monotonic time at which the system could first use a message.
OFI	Order-flow imbalance.
Scene familiarity	Distance or density under a frozen training-scene embedding; not subjective familiarity.
Shadow mode	Live, timestamped prediction with execution disabled.
Paper replica	A simulated policy operating on recorded/live read-only quotes; not a venue fill.
Proper score	A probabilistic scoring rule optimized in expectation by reporting the true belief distribution.

3. The problem: observation is not interpretation

A liquidity heatmap answers a descriptive question: where was displayed size visible through time? HEATWAKE asks a conditional one:

> What state is this liquidity structure in, how is that state changing, what futures remain plausible, and what executable consequence follows?

Static brightness cannot answer that question. The answer, if learnable, lies in the sequence of additions, removals, executions, observed age, approach, replenishment, spread, and cross-venue context.

3.1 Why local prediction is plausible but not assumed

HEATWAKE does not assume global market predictability. It targets local microstructure regularities over seconds to minutes. Research has repeatedly found short-horizon statistical predictability in limit-order-book data, while also showing that predictive accuracy does not necessarily translate into actionable transactions [1–6, 11].

That distinction becomes the project's central discipline:

```
visual plausibility
      ≠
forecast accuracy
      ≠
calibrated probability
      ≠
executable edge
      ≠
live profitability
```

Each transition requires a separate test.

3.2 Why the field matters

The deterministic field creates no information absent from its source messages. It changes the inductive bias: time becomes horizontal structure, relative price becomes vertical structure, and depth/change/execution channels become trackable geometry. The test is whether that bias improves out-of-time generalization, calibration, sample efficiency, or robustness under matched budgets.

---

4. Modeling choice: learn before labeling

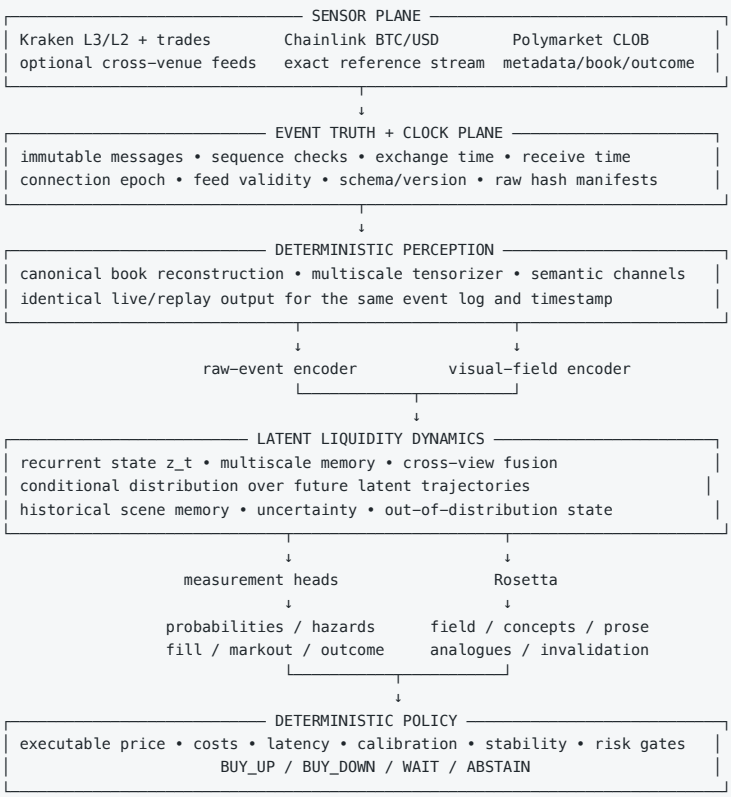
Engineered features and named scenarios remain mandatory baselines, but they should not be the only representation available to the model. HEATWAKE first learns a continuous state from received messages and the semantic field; it then predicts a distribution over future latent trajectories. Human concepts and actions are downstream measurements.

“Thinking in images” therefore does not mean generating a screenshot at every internal step. Events are the record, fields provide spatial structure, latents carry predictive state, and images/language are Rosetta outputs. The full implementation map follows.

---

5. Full system architecture

5.1 System map



5.2 Sensor plane

The first system is deliberately narrow:

1. **Underlying liquidity sensor:** Kraken Spot BTC/USD visible Level 3 plus trades, subject to written rights.
2. **Settlement reference:** directly licensed Chainlink BTC/USD Data Streams, with official Polymarket-resolved outcomes retained as authoritative labels.
3. **Prediction-market observable:** Polymarket market metadata, UP/DOWN Level 2 book, price changes, and resolved outcome, only under permitted read/data use.
4. **Optional cross-venue context:** one or more additional public/licensed BTC books to measure confirmation, divergence, and venue-specific artifacts.

No Coinbase source or dependency is proposed. This is a project constraint, not a comparative claim about Coinbase market-data quality.

5.3 Event truth and clock plane

Every message is stored with:

- Venue and instrument.
- Exchange timestamp as supplied by the venue.
- Local UTC receive timestamp.

- Local monotonic availability time.
- Clock-synchronization uncertainty.
- Sequence/update identifier where supplied.
- Connection epoch and reconnect reason.
- Original message and batch boundary.
- Message type and schema version.
- Raw payload hash.
- Feed-validity state.

Headline and executable evaluation use only information locally available by decision time. If message (e_i) becomes available at local monotonic time (r_i), then:

$$H_t^{avail} = \{e_i : r_i \le t\}.$$

Cross-feed joins are as-of joins on availability time, not exchange timestamp. Exchange-time alignment may be reported only as a separate oracle experiment.

Integrity rules are provider-specific. A sequence discontinuity where sequencing exists, a checksum failure where checksums exist, a clock fault, stale stream, depth reset, or uncertain reconstruction invalidates the interval and forces live abstention. Kraken Level 3 does not supply a global sequence field, so the system must not describe its reconstruction as sequence-complete.

5.4 Deterministic perception

The canonical book engine applies messages in received order within each feed's documented semantics and subscribed depth. It then produces multiresolution numeric tensors—not RGB screen captures.

For a versioned renderer (R_\nu):

$$X_t = R_\nu(H_{t-W:t}^{avail}),$$

where (H_{t-W:t}^{avail}) is the availability-time causal history at time (t). Renderer version, crop, price normalization, time bins, intensity transform, and venue alignment are part of the sample manifest.

The same event log and timestamp must produce the same tensor hash in offline replay and live operation.

5.5 Dual-view encoders

The raw-message view preserves received order, message/batch boundaries, and local inter-arrival timing. It does not claim access to a stronger exchange sequence than the venue provides. The field view exposes spatiotemporal geometry. Each can fail differently, so both must exist as an independent baseline.

The fused state update is:

$$z_t = F_\theta(z_{t-1}, E_t, X_t, C_t),$$

where:

- (E_t) is the structured event representation.
- (X_t) is the multichannel field.

- C_t is causal context such as time remaining, spread, reference distance, and feed health.
- z_t is the latent liquidity state.

The fusion model survives only if it improves locked future data after controlling for parameter count and training compute.

6. What the machine sees

6.1 Raw event schema

The event representation should include, where genuinely available:

FIELD	PURPOSE
Event type	Add, modify, delete, trade, snapshot, status, reconnect.
Side	Bid, ask, aggressive buy, aggressive sell.
Relative price	Ticks or trailing causal normalization around a reference.
Size	Raw and causally normalized quantity.
Inter-arrival time	Event-time rhythm and bursts.
Exchange/receive/availability time	Market payload timing, local latency, and the causal decision boundary.
Sequence/checksum	Reconstruction integrity where the venue supplies it.
Order ID	Visible-order lifecycle only when true L3/MBO supplies it.
Age mask	Age since first local observation and last amendment, with left-censoring explicitly marked.
Feed mask	Stale, gapped, degraded, or healthy.

An order ID identifies an order, not a person. HEATWAKE does not infer participant identity from Kraken order IDs.

6.2 Semantic field channels

The first field contains:

1. Resting bid depth.
2. Resting ask depth.
3. Bid additions.
4. Ask additions.
5. Bid removals, with cancellation/execution attribution only where validated.
6. Ask removals, with cancellation/execution attribution only where validated.
7. Aggressive buy executions.
8. Aggressive sell executions.
9. Age since local observation and last amendment, plus censoring masks.
10. Replenishment and depletion.
11. Midpoint, best bid/ask, and spread.

12. Feed validity and staleness.

Later channels may include:

- Cross-venue depth and basis.
- Price-to-beat distance.
- Chainlink update age.
- Polymarket UP/DOWN depth and quoted prices.
- Time remaining to settlement.
- Venue disagreement and clock skew.

6.3 Multiple clocks and resolutions

Markets do not evolve on one clock. The model should view:

- **Message/event time:** received message and batch order, plus exchange timestamps as attributes.
- **Micro time:** tens to hundreds of milliseconds.
- **Fast context:** seconds.
- **Structural context:** minutes.
- **Contract time:** fraction of the 5- or 15-minute settlement window remaining.

The raw tape is stored once. Fields at different cadences are rendered later. Taking 1,000 screenshots per second would create redundant, lossy, UI-dependent data and still miss the distinction between multiple events within a frame.

6.4 Causal normalization

All scaling must use only information available by time (t). Prohibited examples include:

- Full-day maximum depth.
- Future volatility.
- A color scale fitted on the completed clip.
- Centering a crop using the future price path.
- Selecting a wall because it later mattered.

The renderer is part of the model and is therefore versioned, tested, and ablated.

---

7. The Liquidity Wavefunction

7.1 Operational definition

The Liquidity Wavefunction is the product name for a rolling conditional distribution over future latent liquidity trajectories:

$$\mathcal{B}_t = p_\theta(Z_{t+1:t+H} \mid H_t^{avail}),$$

where (H_t^{avail}) contains only messages locally available by time (t). It may be represented by a mixture distribution, conditional flow, diffusion samples, or an ensemble. The first implementation should use the simplest distribution that is calibrated on decoded observables and computationally tractable.

One concrete representation is:

$$\mathcal{B}_t = \{(\tau^{(k)}, w_t^{(k)})\}_{k=1}^{K}, \qquad w_t^{(k)} \geq 0, \quad \sum_{k=1}^{K} w_t^{(k)} = 1,$$

where each (\tau^{(k)}) is a coherent future latent trajectory and (w_t^{(k)}) is its current normalized weight.

Unless a transition model, observation likelihood, and explicit inference procedure are implemented, (\mathcal{B}_t) is a conditional ensemble recomputed or recurrently updated at each time—not a Bayesian posterior or particle filter. New evidence may concentrate, disperse, or split probability mass, and the forecast horizon rolls forward.

7.2 Why multiple futures must survive

A single expected future field can average incompatible modes into fiction. If half the probability mass predicts a sharp rejection and half predicts a clean breach, their pixel average may show a blurry path that is not itself plausible.

The model must preserve:

- Multiple coherent futures.
- Probability mass by horizon.
- Dependencies between liquidity and price paths.
- Mode splitting under uncertain evidence.
- Increasing uncertainty where the future is not identifiable.

The model's job is not to force a signal. Its job is to represent uncertainty accurately enough that policy eligibility is rare and meaningful.

7.3 Market “superposition”

The word superposition is permitted only in a carefully bounded sense:

- A high-dimensional classical representation can encode several candidate states at once.
- A probabilistic model can preserve several future modes before readout.
- A linear latent representation may combine features or hypotheses.

This is a classical high-dimensional superposition analogy, not quantum superposition. It establishes no entanglement, quantum parallelism, or computational advantage.

7.4 Non-core amplitude experiment

After the real-valued model is established, an experimental tensor-network or complex-amplitude head may represent:

$$\psi_t(s) = \sum_j a_{j,s} e^{i\phi_{j,s}}, \qquad P_t(s) = \frac{|\psi_t(s)|^2}{\sum_{s' \in \mathcal{S}} |\psi_t(s')|^2}, \qquad s \in \mathcal{S}.$$

Here (\mathcal{S}) must be a defined discrete outcome basis. This experiment belongs outside the core product path and is retained only if it improves proper probabilistic scores, robustness, or parameter efficiency against matched real-valued density heads.

8. Liquidity structures as a multistate process

The intuitive categories HOLD, PULL, ABSORB, and BREAK are useful interface summaries but are not mutually exclusive terminal labels. A structure may partially pull and later trade; it may absorb for several seconds and then breach.

For a structure selected using information available at anchor time (t_0), track a multistate path such as:

```
UNTOUCHED
   ├─→ WEAKENED / PARTIALLY REMOVED
   ├─→ TOUCHED
   │       ├─→ ABSORBING
   │       │       ├─→ REJECTED
   │       │       └─→ BREACHED
   │       └─→ BREACHED
   └─→ EXPIRED / ADMINISTRATIVELY CENSORED
```

The model estimates allowed transition intensities:

$$\lambda_{ij}(\tau \mid z_t), \qquad i \to j \text{ allowed by the state graph.}$$

Where a first-transition competing-risk benchmark is used, cumulative incidence is:

$$F_{ij}(\tau) = \int_0^\tau S_i(u)\lambda_{ij}(u)\,du,$$

with the risk set and survival term (S_i) defined before evaluation. For the first benchmark, HEATWAKE may simplify to mutually exclusive first qualifying transitions or fixed landmark states, but that simplification must be explicit.

The rules for structure selection, merge/split tracking, touch, material removal, replenishment, rejection, breach, dwell time, and administrative censoring must be frozen using training data before the final test. “HOLD” means no qualifying transition by the declared horizon; it is not a permanent terminal state.

HEATWAKE may describe a pattern as “spoof-like pulling” for annotation purposes. It may not infer manipulation or intent from order behavior alone.

9. Rosetta: observability without forced serialization

9.1 Purpose

Rosetta exists because an opaque latent model is difficult to debug, evaluate, or use responsibly. It is trained and evaluated as a separate subsystem.

Forecast ownership remains in the core measurement heads. Rosetta may display a core probability or future-field decoder output, but it may not generate a second, untracked forecast. Post-hoc probes are labeled as probes; they are not silently promoted to model reasoning.

Rosetta produces five classes of output:

1. **Visual readout:** observed field, future-field haze, coherent trajectory samples, concentration ridges, and fragility zones.
2. **Concept readout:** pull hazard, absorption hazard, breach hazard, replenishment, execution pressure, novelty, and feed health.
3. **Historical readout:** nearest prior scenes, each with one realized continuation; the neighbor set induces an empirical outcome distribution.
4. **Sensitivity/intervention readout:** how the model output changes under valid input perturbations or on-manifold latent interventions.
5. **Language readout:** a constrained explanation of the above evidence.

9.2 Rosetta is not a perfect transcript

A learned decoder cannot be assumed to reconstruct every nuance of a continuous latent state. High probe accuracy does not prove that the model actually uses the decoded concept. Pretty saliency maps do not prove faithfulness [23–26].

Accordingly, Rosetta has three confidence levels:

LEVEL	OUTPUT	CLAIM
Provenance	Exact input events, timestamps, renderer version, model version, policy version.	Deterministic audit fact.
Measured diagnostic	Probabilities, sampled futures, concept heads, neighbor outcomes, ablation deltas.	Model-derived and quantitatively testable.
Narrative gloss	Natural-language explanation.	Human summary; never independent evidence.

The language model, if one is used at all, receives only structured diagnostics. It may phrase them; it may not invent a condition, causal story, probability, or trade.

9.3 Faithfulness tests

Rosetta is evaluated on:

- Reconstruction error for observed and future fields.
- Calibration of concept probabilities.
- Stability under irrelevant visual changes.
- Sensitivity to relevant, availability-time input perturbations; these are model-sensitivity results, not market-causal claims.
- On-manifold activation patching or mediation tests where feasible; these support only model-internal causal claims.
- Probe selectivity: a concept that can be decoded but is not used by the forecast must be labeled diagnostic-only.
- Predictive equivalence: the displayed distribution must match the measurement head.
- Explanation consistency across identical inputs and versions.
- Explicit “unknown” output when the decoder cannot support a gloss.

The doctrine is:

> **Observability is permanent, separately tested, and never treated as independent evidence.**

10. The Bitcoin and prediction-market experiment

10.1 Why Bitcoin first

Bitcoin is the proposed engineering pilot because:

- It trades continuously.
- It permits unattended data capture outside the conventional workday.
- Multiple independent venues expose live order-book data.
- Recurring short-duration binary markets create explicit settlement events.
- The underlying price, prediction-market quote state, and market depth can be observed simultaneously.

This makes BTC a useful laboratory. It does not imply that crypto microstructure is easier, stationary, or universally transferable.

10.2 Recommended non-Coinbase sensor

The preferred first underlying sensor is **Kraken Spot BTC/USD visible Level 3 plus the trade channel**, subject to written permission for archival, machine-learning, and derived-data use.

Kraken's official Level 3 WebSocket documentation describes individual visible orders with order IDs, limit prices, remaining quantities, timestamps, and add/modify/delete updates, with subscribed depth options of 10, 100, or 1,000 price levels [35–37]. This supports depth-bounded visible-order tracking that an aggregated Level 2 feed cannot provide.

Important limits remain:

- Hidden iceberg quantity is not visible.
- In-flight or unmatched market orders are not visible.
- Untriggered stops are not visible.
- Orders that fall outside the subscribed depth may disappear without an explicit delete.
- Snapshot timestamps reflect insertion or amendment, so original order age can be left-censored.
- A delete may represent cancellation or full fill; the trade stream can aid attribution but does not guarantee perfect classification.
- The feed has no global sequence field; integrity relies on documented message handling, depth resets, and available checks.
- The available checksum covers top-of-book levels rather than proving completeness of every subscribed level.
- An order ID is not a trader identity.
- The official historical download is time-and-sales, not a ready-made historical L3 corpus.

Therefore the dataset must be collected forward or obtained under a separate archive agreement [53–54].

10.3 Chainlink is the settlement reference

Recurring international Polymarket BTC Up/Down markets resolve against Chainlink BTC/USD, not Kraken, Coinbase, a Binance candle, or the HEATWAKE midpoint [42–45].

For the documented 5- and 15-minute contracts:

- UP wins if the ending Chainlink BTC/USD value is greater than or equal to the starting value.
- DOWN wins otherwise.
- Equality belongs to UP.
- The opening Chainlink value is the price to beat.

HEATWAKE therefore needs two conceptually different prices:

1. **Underlying liquidity sensor:** the venue books that may contain predictive microstructure.
2. **Settlement oracle:** the exact Chainlink reference that determines the contract outcome.

The model must learn the relationship between them without confusing one for the other.

10.4 Polymarket is a second observable

The Polymarket book contains quoted state prices and its own microstructure. Bid and ask quotes may be transformed into market-implied probabilities for analysis, but they are not themselves calibrated probabilities. Polymarket is not the source of Bitcoin reality.

When permitted, the system records:

- Market slug and condition/token IDs.
- Window start/end.
- Official price to beat.
- UP and DOWN Level 2 snapshots.
- Price-level updates.
- Best executable bid and ask.
- Trades and official resolution.
- Tick-size and market-state changes.
- Time remaining.

Polymarket's public WebSocket feed is aggregated Level 2. It does not expose individual orders or participant identity [38–40].

Polymarket CLOB V2 launched on April 28, 2026 with new exchange contracts, pUSD denomination, event/schema changes, and match-time fee mechanics [62–63]. Every record must therefore preserve the protocol version, exchange contract, stream schema, denomination, market rules, and fee schedule in effect at the decision time. UP and DOWN tokens are complementary claims and must be modeled jointly, including mint/merge relationships and no-arbitrage checks, rather than treated as unrelated books.

A 2026 preprint studying pre-cutover Polymarket microstructure reports that trade direction inferred from the public book feed agreed poorly with authoritative on-chain fills [13]. That result must not be generalized to V2 without replication. HEATWAKE should join the current V2 OrderFilled and OrdersMatched records when direction-dependent microstructure is studied.

10.5 Current contract scope

As of July 10, 2026, official Polymarket sources show recurring BTC Up/Down **5-minute** and **15-minute** series. A live official-series query for a 30-minute slug returned no result on that date; this is not proof that such a series never existed or will not launch. The response must be archived with the research snapshot before making an absence claim. Contract availability and rules are discovered dynamically rather than hard-coded.

The proposed sequence is:

1. Five-minute contracts for faster research feedback.
2. Fifteen-minute contracts for the primary product demonstration.
3. Any later horizon only after official discovery and rule verification.

10.6 Geographic and legal boundary

The international Polymarket CLOB currently lists the United States as a blocked jurisdiction for order placement [41, 55]. The project will not route around or attempt to circumvent that restriction.

For a New York-based research program:

- International Polymarket may be used only for lawful read-only research under applicable data terms.
- Live orders remain disabled.
- The initial product is a paper replica and shadow evaluator.
- Polymarket US is monitored as a separate regulated venue [56–58].
- A dated public-catalog search on the paper date did not show equivalent recurring 5- or 15-minute BTC direction products; the response must be archived before publication and rechecked dynamically.

If a lawful venue later lists an equivalent product, the execution adapter can change without retraining the entire perception model, but venue-specific calibration and execution testing remain mandatory.

10.7 The central benchmark

For a binary market with model settlement probability (p_t), the relevant baseline is not 50%. It is the size-dependent market quote available after latency and fees.

Headline forecasts are evaluated at predeclared contract checkpoints, such as fixed values of time remaining. A continuously updating interface may be studied separately, but thousands of updates sharing one settlement label are not independent trials.

The first economic benchmark is intentionally simple:

- One decision at a declared checkpoint.
- Fixed small quantity (q).
- All-or-none taker fill-or-kill (FOK) marketable-limit semantics where supported.
- No maker-fill claim from public Level 2 data.
- No partial-fill accounting in the first benchmark: the full (q) fills or the result is no trade.
- No pre-existing inventory.
- Hold a filled binary claim to settlement.
- Actual venue fee rules and token complementarity.

For candidate action (a\in{\text{BUY_UP},\text{BUY_DOWN}}), let:

- (t+\ell) be the modeled venue-arrival time.
- (\widehat K_t^{FOK}(a,q,\ell)) be a conservative pretrade estimate of the per-share full-fill cost after latency, including the venue's nonlinear fee rule.
- (\mathcal{P}_t) be a predeclared forecast set produced by a time-dependent calibration procedure. Its empirical coverage is reported by contract/day stratum; it is not claimed to be a pointwise guarantee.
- (\pi_{\text{BUY_UP}}(p)=p) and (\pi_{\text{BUY_DOWN}}(p)=1-p).
- (\delta>0) be a safety margin selected on the policy-validation period.

A conservative filled-position value is:

$$\underline{EV}_t(a,q) = q \inf_{p\in\mathcal{P}_t} \left[\pi_a(p) - \widehat{K}_t^{FOK}(a,q,\ell)\right].$$

Replay walks the actual arrival-time book: the FOK order either fills all (q) at its realized cost or produces zero position and zero P&L. Live operation would require a separately validated arrival-cost model. FAK, partial fills, inventory, early exit, and maker execution each require a new policy protocol. If several actions clear (\delta), the policy chooses the highest lower-bound value subject to exposure rules. If no action clears but data and risk gates are valid, the result is WAIT. If a data, model-support, legal, or risk gate fails, the result is ABSTAIN.

The market-implied state is both an input and a baseline. Evaluation therefore separates an underlying-only model, a Polymarket-only model, and the fused model. A model that cannot improve on the market-only baseline has not earned a prediction-market product.

11. Outputs and the meaning of conviction

11.1 Forecast outputs

LWM-1 should produce distributions, not slogans:

- Future latent liquidity-field distribution at multiple horizons.
- Pull-before-touch hazard.
- Absorption/touch hazard.
- Breach hazard.
- Time to liquidity event.
- Future return distribution.
- Maximum favorable and adverse excursion.
- Spread and volatility distribution.
- Barrier-hit probabilities.
- Fill probability under a specified order policy.
- Markout conditional on fill.
- Binary settlement probability where a contract exists.
- Scene familiarity and out-of-distribution score.
- Feed-health and synchronization state.

11.2 Conviction is a gate vector

“Conviction 87%” is not scientifically meaningful by itself. HEATWAKE defines conviction as a vector:

$$\kappa_t = (\text{lower-bound executable value}, \text{validated reliability stratum}, \text{forecast dispersion}, \text{temporal stability}, \text{scene familiarity}, \text{model agreement}, \text{feed health}).$$

Calibration is a property measured over populations or declared strata, not a magic per-instance score. An action is eligible only when its current stratum has passed out-of-time reliability checks and every other required component clears a predeclared threshold. High directional probability with weak reliability evidence, stale data, unfamiliar structure, or no executable value is not conviction.

Temporal stability is the forecast's sensitivity to small availability-time shifts; model agreement is dispersion across frozen ensemble members or checkpoints; scene familiarity is the preregistered embedding-density/distance measure defined in the glossary. Each has an explicit estimator and threshold rather than an interface-only score.

11.3 Deterministic decisions from probabilistic forecasts

The conditional forecaster should remain probabilistic. The policy should be deterministic and auditable.

Given:

- the same immutable event history,
- the same model checkpoint,
- the same calibration artifact,
- the same executable quotes,

- the same policy version, and
- the same random seed when sampling is involved,

HEATWAKE must reproduce the same output and decision.

Determinism belongs at the policy and audit boundary. It does not require pretending the future itself is deterministic.

11.4 WAIT versus ABSTAIN

The system-level default is ABSTAIN until data, model-support, legal, and risk gates pass.

WAIT means the scene is valid and evaluable but no BUY_UP or BUY_DOWN candidate clears the policy threshold. ABSTAIN means the system refuses to evaluate or act because one or more epistemic, operational, legal, or risk gates failed.

---

12. Model and training program

12.1 Start small

The first model is not a frontier-scale foundation model. It is a compact, domain-specific system trained from a new event archive.

The reference implementation, **LWM-1A**, is:

1. A causal temporal-convolution event encoder over received message features.
2. A compact causal 3D-convolution field encoder.
3. Gated recurrent cross-view fusion with explicit data-validity masks.
4. A finite-mixture future-latent head; the number of modes is selected on validation.
5. Multistate transition, path, execution, and settlement heads.
6. A frozen scene embedding for approximate-nearest-neighbor retrieval.

The parameter and latency budgets are frozen after the Phase 0 hardware/data profiler, and comparison models must be matched within the preregistered tolerance. State-space, Transformer, patch-field, conditional-flow, and diffusion variants are challengers—not simultaneous requirements. They replace LWM-1A only if they improve locked validation under the same budget.

Calibration is selected per observable output and time-dependent validation regime. Temperature scaling, isotonic regression, ensembles, or conformal-style procedures are candidates; none is assumed to produce pointwise probability guarantees [27–29].

12.2 Training curriculum

Stage A — reconstruction and integrity

Before modeling:

- Rebuild the book from snapshots and deltas.
- Validate provider-specific order, depth, reset, and checksum semantics.
- Prove deterministic live/replay tensors.
- Measure availability-time latency, clock-sync uncertainty, reconnect behavior, and data loss.
- Freeze a causal sample manifest.

No predictive result is meaningful until this stage passes.

Stage B — self-supervised perception

Train the encoders to learn market structure without trading labels:

- Masked event prediction.
- Masked field-channel prediction.
- Next-event type, side, relative price, size, and time.
- Future latent-field prediction.
- Cross-view consistency between the event and field representations.
- Temporal contrast between genuinely nearby and unrelated scenes.

The goal is not to reconstruct every pixel. It is to retain information useful for future dynamics.

Stage C — liquidity lifecycle

Train multistate or explicitly simplified first-transition heads for:

- Material withdrawal before touch.
- Touch.
- Absorbing behavior.
- Rejection.
- Persistent breach.
- Administrative censoring and time in state.

These labels are generated objectively from future events using thresholds fixed before the final test.

Stage D — path and execution

Add:

- Multi-horizon return distributions.
- Barrier probabilities.
- Favorable/adverse excursion.
- Spread and volatility.
- Fill probability.
- Markout conditional on fill.

Maker and taker targets are separate because their economics and adverse selection differ.

Stage E — prediction-market fusion

Add the exact contract state:

- Chainlink price-to-beat distance.
- Time remaining.
- Chainlink update age.
- UP/DOWN executable book.

- Current market-implied quote/book state and liquidity.
- Official settlement label.

The underlying model should still be evaluable without Polymarket inputs so that the incremental value of the prediction-market book can be measured.

Stage F — Rosetta

Train and test:

- Future-field decoder.
- Probabilistic concept layer.
- Historical scene retrieval.
- Input-sensitivity and on-manifold latent-intervention tests.
- Constrained language renderer.

Rosetta is trained against a frozen core or through a stop-gradient boundary. It is not allowed to improve headline forecasts by leaking human annotations or future information into the predictive representation. Joint training is a separately reported ablation, never the default.

Stage G — policy

Freeze the predictive model first. Then select a deterministic action policy on a separate validation period. Do not begin with reinforcement learning.

Offline reinforcement learning or world-model planning is considered only after:

- the simulator is validated,
- action-conditioned market response is modeled,
- historical policy bias is understood, and
- supervised execution-aware baselines are exhausted.

12.3 Loss design

For horizon (h), LWM-1A uses an exponential-moving-average target encoder (g_{\bar\theta}) with no gradient into the target branch:

$$y_{t+h} = \operatorname{stopgrad}\left(g_{\bar\theta}(E_{t+h}, X_{t+h})\right), \qquad \hat{y}_{t+h} = P_\theta(z_t, h).$$

The future-latent loss combines normalized feature prediction, variance/covariance anti-collapse regularization, and a proper decoded-observable likelihood:

$$\mathcal{L}_{future\ latent} = \left\| \frac{\hat{y}_{t+h}}{\|\hat{y}_{t+h}\|_2} - \frac{y_{t+h}}{\|y_{t+h}\|_2} \right\|_2^2 + \beta \mathcal{L}_{var/cov} - \gamma \log p_\theta(Y_{t+h} \mid z_t),$$

where (Y_{t+h}) is a declared future observable such as depth/change/execution channels. Future inputs appear only in the training target. A decoded-likelihood-only variant and a no-future-objective model are mandatory ablations.

A possible multi-task objective is:

$$\mathcal{L} = \lambda_{latent}\mathcal{L}_{future\ latent} + \lambda_{event}\mathcal{L}_{next\ event} + \lambda_{state}\mathcal{L}_{multistate} + \lambda_{path}\mathcal{L}_{path} + \lambda_{exec}\mathcal{L}_{execution} + \lambda_{settle}\mathcal{L}_{settlement}.$$

Loss weights are selected without opening the final test. Every head must demonstrate incremental value; multi-task complexity is not automatically beneficial.

After the core checkpoint (\theta^*) is frozen, Rosetta is optimized separately:

$$\mathcal{L}_{Rosetta} = \mathcal{L}_{field\ readout} + \mathcal{L}_{concept} + \mathcal{L}_{faithfulness}, \qquad \text{stopgrad}(\theta^{\backslash *}).$$

12.4 Historical market memory

Each scene receives a normalized embedding and a strict timestamp. Retrieval at time (t) may search only episodes that ended before (t).

The returned record includes:

- Similarity and distance.
- Venue and regime.
- Its one realized forward liquidity-field trajectory.
- Structural events.
- Executable forward outcomes.
- Data-quality state.

The neighbor set, not an individual scene, induces an empirical forward distribution. Retrieval is valuable even if directional alpha fails: it can support replay search, expert review, anomaly detection, and education. Those uses must not be confused with profitability.

13. The scientific contract

13.1 Research ladder

HEATWAKE must climb this ladder in order:

GATE	QUESTION	EVIDENCE REQUIRED
G0 — Integrity	Is the depth-bounded reconstructed market consistent with provider semantics?	Provider-specific checks, availability-time replay, deterministic tensor hashes, explicit uncertainty masks.
G1 — Representation	Does the field preserve useful dynamics?	Field model beats trivial/engineered baselines on proper scores.
G2 — Fusion	Does the field's inductive bias improve learning?	Fused model beats event-only and field-only under matched data, parameters, compute, and latency.
G3 — Futures	Does latent future prediction improve downstream forecasts?	Ablation with and without the future objective/decoder.
G4 — Calibration	Do probabilities mean what they say?	Reliability, proper scoring rules, risk-coverage behavior.
G5 — Market edge	Does the model beat the market-only quote baseline?	Frozen chronological test using size-dependent arrival quotes and current fees.
G6 — Execution	Can the edge be captured?	Pessimistic taker arrival/fill replay first; maker queue claims require stronger data.
G7 — Live shadow	Does it survive current data?	Timestamped predictions committed before outcomes.
G8 — Capital	Is live deployment legal and justified?	Rights, eligibility, operational controls, and untouched live evidence.

Failure at a gate prevents claims above that gate.

13.2 Required baselines

Every baseline receives identical source data, contexts, labels, splits, and policy rules.

Statistical and microstructure baselines

- Class prior and persistence.
- No-change.
- Momentum and mean reversion.
- Microprice.
- Queue/depth imbalance.
- Linear/logistic order-flow imbalance.
- Queue-reactive model.
- State-dependent Hawkes-style model where feasible.

Conventional machine learning

- Logistic regression.
- Gradient-boosted trees on causal engineered features.
- Temporal convolution or recurrent model on stationary order-flow features.

Deep LOB and representation

- DeepLOB-style model.
- Raw-event causal model.
- Single-frame field model.
- Field-video model.
- Event-only model.
- Field-only model.
- Fused HEATWAKE model.

Generative models

- Conditional persistence.
- Simple mixture-density future head.
- Autoregressive LOB generator.
- Structured-image diffusion/inpainting baseline.
- Relevant LOB-Bench reference models.

Trading baselines

- No trade.
- Contemporaneous market quotes and spread.
- OFI threshold.
- Microprice threshold.

- Simple momentum and mean reversion.
- The identical execution policy driven by each predictive baseline.

Prediction-market baselines

- Chainlink distance-to-threshold, time remaining, and causal volatility in a digital-option baseline.
- Polymarket-only model using contemporaneous book state, spread, time remaining, protocol version, and fee state.
- Underlying-only HEATWAKE model with no Polymarket features.
- Fused or residual model that must improve paired proper scores over the Polymarket-only model.
- UP/DOWN complement, mint/merge, and no-arbitrage checks.

Published benchmark scores do not substitute for retraining on the HEATWAKE corpus.

13.3 Mandatory ablations

- RGB screenshot versus semantic numeric field.
- Event-only versus field-only versus fusion.
- Remove depth, adds, removals, trades, observed-age, replenishment, and feed-health channel groups.
- Level 2 versus genuine Level 3/MBO.
- Clock-time versus event-time sampling.
- Single-resolution versus multiscale history.
- Single-horizon versus joint multi-horizon training.
- With and without future-latent objective.
- With and without self-supervised pretraining.
- Deterministic versus stochastic future rollout.
- With and without cross-venue context.
- With and without Polymarket book state.
- With and without calibration/abstention.
- Asset-specific versus pooled model.
- Classical density head versus tensor-network/Born head.
- Synthetic augmentation versus real-only training.

Negative controls are mandatory:

- Shift labels.
- Permute time.
- Permute relative price.
- Break chronology deliberately.
- Randomize renderer colors while preserving numeric channels.

If a supposedly predictive model survives broken chronology or shifted labels, the pipeline is leaking.

13.4 Chronological evaluation

Random screenshot splits are prohibited.

The dataset is divided by complete chronological blocks:

- Training period.
- Tuning/early-stopping period.
- Calibration period.
- Policy-selection period.
- Untouched final test.
- Later live shadow period.

Overlapping contexts and label horizons are purged across boundaries. Embargoes prevent near-duplicate episodes from straddling splits. The final test is opened once after architecture, thresholds, and policy are frozen.

For prediction markets, the primary unit of inference is the contract at a predeclared decision checkpoint. If several checkpoints per contract are retained, score differences are paired and clustered by contract and day. Adjacent contracts and repeated states are not treated as independent observations. Use moving-block or heteroskedasticity/autocorrelation-consistent inference where appropriate, and correct for the number of tried model/policy families.

Additional holdouts should include:

- Volatility regimes.
- Venue changes.
- Feed degradation.
- Major events.
- Different times of day/week.
- A later market period.
- A second instrument or venue when data supports it.

13.5 Forecast scoreboard

Report:

- Negative log likelihood.
- Brier score.
- Continuous ranked probability score or energy score.
- Calibration error and reliability diagrams.
- Area under the precision–recall curve (AUPRC) and Matthews correlation coefficient (MCC) for rare structural events.
- Time-dependent multistate or cumulative-incidence metrics.
- Precision versus coverage.
- Complete risk-coverage curve.
- Out-of-distribution detection.
- Multi-step rollout degradation.
- Performance by horizon, venue, regime, volatility, and time remaining.
- Future-field distribution metrics rather than screenshot similarity alone.

13.6 Trading scoreboard

Report separately:

- Fill probability.
- Markout conditional on fill.
- Adverse selection.

- Spread captured or paid.
- Fees and rebates.
- Slippage and partial fills.
- Decision-to-arrival latency.
- Turnover.
- Net expectancy.
- Drawdown.
- Deflated Sharpe ratio.
- Probability of backtest overfitting.
- Block-bootstrap confidence intervals.
- Live shadow degradation relative to replay.

The first binary-market policy is the one-shot, fixed-quantity, arrival-book policy defined in Section 10.7. Maker and multi-position strategies require a new protocol rather than a silent extension of the first result. Prediction metrics and trading metrics remain two separate scoreboards.

13.7 Pre-registration

Before opening the final test, freeze:

- Dataset fingerprint.
- Renderer and label versions.
- Model family and parameter budget.
- Baselines.
- Hyperparameter search budget.
- Metrics.
- Calibration method.
- Action thresholds.
- Cost and latency assumptions.
- Kill criteria.

The initial pilot estimates variance, serial dependence, event prevalence, and data loss. A power analysis then freezes the minimum effect size, confidence level, required contracts/days/regimes, minimum coverage, calibration tolerance, and maximum acceptable live-shadow degradation. If the study cannot be powered within the licensed corpus, the final test is deferred rather than weakened.

The timestamped protocol should be public or independently witnessed before results are generated.

---

14. Failure modes and kill criteria

14.1 Data and representation failures

- Sequence gaps are silently accepted.
- Snapshot and delta semantics diverge.
- Exchange time is confused with receive time.

- Cross-venue clock skew creates artificial lead/lag.
- Renderer scaling uses future information.
- Overlapping windows leak across splits.
- The model learns venue/session artifacts.
- Order IDs are treated as identities.
- Level 2 data is described as MBO.
- Synthetic samples contaminate the real test set.

14.2 Modeling failures

- High accuracy comes from the dominant no-change class.
- A future image looks coherent but misses local executable detail.
- A stochastic model averages or ignores important modes.
- A concept head is readable but harms the core forecast.
- A retrieval index contains future episodes.
- Class rebalancing destroys probability calibration.
- A latent decoder tells a persuasive but causally false story.
- Regime-specific gains are presented as universal.

14.3 Execution failures

- Midpoint direction is treated as tradable P&L.
- The backtest fills passive orders exactly when adverse selection is highest.
- Queue position is ignored.
- Historical liquidity is reused after a simulated fill.
- Exit cost is omitted.
- Decision-time state substitutes for arrival-time state.
- Paper fills are called live fills.
- Many strategy trials are hidden behind one reported Sharpe ratio.

14.4 Adversarial market failures

- Displayed liquidity is intentionally transient.
- The model becomes predictable to counterparties.
- Venue rules or tick sizes change.
- Latency shifts erase a lead.
- A market maker changes behavior after the model is deployed.
- Cross-venue hedging makes a local book misleading.
- Distribution drift occurs faster than retraining and validation.

14.5 Kill criteria

The relevant branch is removed or the product is not shipped when:

1. Field inputs add no robust information beyond event-only baselines.
2. Latent future modeling adds no robust information beyond direct discriminative heads.

3. Rosetta explanations fail faithfulness tests.
4. The optional amplitude/tensor-network head does not beat a matched classical head.
5. The model fails calibration or selective-risk tests on the untouched future.
6. Statistical advantage disappears at executable quotes after realistic costs.
7. Shadow results materially decay relative to replay.
8. Required data, training, or display rights cannot be secured.
9. The intended execution venue is not legally available.

An unsuccessful directional result may still leave valuable replay search, annotation, anomaly detection, or visualization technology. Those become separate products with separate claims.

---

15. The visual product

15.1 The interface should expose the field

The primary surface is not a dashboard full of indicators. It is the observed liquidity field continuing through a visible NOW boundary into the model's conditional future field.

The interface may show:

- Resting-liquidity history behind NOW.
- Live additions, removals, and executions.
- Future-field haze ahead of NOW.
- Multiple translucent coherent trajectories.
- Concentration ridges where many futures overlap.
- Dark zones where new evidence has depleted probability mass.
- Fragility contours where the distribution may split.
- Pull, absorption, and breach hazard at selected structures.
- Prediction-market probability and executable spread.
- Model/market disagreement.
- Data health, uncertainty, novelty, and abstention.

A bright directional wake appears only when the field is concentrated, calibrated, stable, familiar, healthy, and executable.

Every visual encoding must have a quantitative legend and versioned mapping. Usability tests must measure whether users over-trust the future field, mistake samples for certainty, or ignore abstention. A visually compelling overlay that worsens decisions fails the product test even if its underlying forecast is unchanged.

15.2 The zoom demonstration

The site should let a viewer zoom from the complete system into the exact evidence:

```
LEVEL 1 — THE MARKET
Combined order-flow heatmap and future field.

LEVEL 2 — THE STRUCTURE
Select a wall or liquidity ridge; inspect observed age/censoring, additions, removals, trades,
replenishment, approach, and cross-venue confirmation.

LEVEL 3 — THE MODEL STATE
Inspect sampled future modes, probability mass, uncertainty, and nearest scenes.

LEVEL 4 — ROSETTA
See which concepts are measured, what interventions changed the output,
and what evidence would invalidate the current reading.

LEVEL 5 — THE POLICY
See executable edge, costs, risk gates, and why the result is
BUY_UP, BUY_DOWN, WAIT, or ABSTAIN.
```

The demonstration should make the architecture understandable without pretending that the latent state is fully human-readable.

15.3 The simulated demo contract

Before a validated model exists, the public demonstration remains simulated and visibly labeled.

The proposed demo boundary is:

```
seeded synthetic event tape
        ↓
real deterministic book reconstruction and tensorization
        ↓
mock model response matching the future API contract
        ↓
real policy and Rosetta interface logic
        ↓
interactive field
```

Permanent labels:

```
DATA        SYNTHETIC
TENSORIZER  IMPLEMENTATION STATUS SHOWN
MODEL       MOCK RESPONSE
EXECUTION   DISABLED
RESULTS     NOT VALIDATED
```

Synthetic data never enters the real training or test corpus. The demo is a specification of the product vision, not evidence of prediction.

15.4 What a validated signal must show

Every signal includes:

- Instrument and contract.
- Timestamp and horizon.
- Model checkpoint and policy version.
- BUY_UP/BUY_DOWN/WAIT/ABSTAIN.
- Settlement probability, forecast set, and reliability stratum.
- Executable bid/ask used.

- Expected edge after costs.
- Coverage/familiarity state.
- Feed-health state.
- Dominant measured conditions.
- Invalidation conditions.
- Nearest historical scenes.
- A link to the committed forecast record.

Every signal must carry this provenance.

---

16. Proposed implementation stack

This is an engineering reference architecture, not a claim that the system is already implemented.

BOUNDARY	PROPOSED TOOL
Direct feed capture	Rust with Tokio/WebSocket adapters
Optional Bookmap integration	Java 21 + Gradle + Bookmap Layer 1 API
Cross-language schema	Protocol Buffers
Local service transport	gRPC
Canonical book/tensor engine	Shared Rust core
Rust to Python	PyO3 + Maturin
Rust to browser	WebAssembly
Immutable event storage	Apache Arrow/Parquet, hourly partitions, hash manifests
Dense tensor shards	Zarr
Local analysis	Polars + DuckDB
Training	Python 3.12 + pinned PyTorch
Apple acceleration	PyTorch MPS
Experiment tracking	MLflow
Dataset/checkpoint versioning	DVC plus content hashes
Configuration	Hydra or equivalent immutable configs
Replay/backtesting	NautilusTrader plus custom prediction-market execution logic
Inference	FastAPI/Uvicorn or a lower-latency service after profiling
Live UI transport	WebSocket
Frontend	Existing React + TypeScript + Vite
Field renderer	PixiJS/WebGL
Public hosting	Existing Vercel project
Optional burst GPU	Modal
Optional object storage	Cloudflare R2 or equivalent

16.1 Why a shared core matters

Book reconstruction, provider-specific integrity validation, price/time binning, semantic channels, causal normalization, and structure labels must not be separately reimplemented in Python, TypeScript, and Java.

A shared deterministic core should power:

- Offline dataset generation.
- Live inference.
- Browser simulation.
- Replay.
- Tests.

The browser's synthetic demo and the research pipeline may have different data sources, but they should share the same event and tensor contracts.

16.2 No LLM credit dependency

The core model is custom PyTorch inference. It does not call OpenAI, Anthropic, or another hosted LLM for market prediction.

There are therefore no token-credit costs in the decision loop.

An optional language model may phrase Rosetta diagnostics for a research report or interface, but:

- it is not required for prediction,
- it receives constrained structured evidence,
- it cannot place an order,
- it can be local,
- and the system remains functional without it.

16.3 Expected M4 Max scope

An M4 Max is expected to handle the following, subject to profiling against the exact memory configuration, feed rate, tensor resolution, and model:

- Continuous data capture.
- Book reconstruction and tensor generation.
- Local analytics and replay.
- The simulated site.
- Small and medium perception models.
- Baseline training and ablations.
- LWM-1 inference.
- Initial latent dynamics experiments.

Apple supports GPU-accelerated PyTorch through the MPS backend [48–49]. Phase 0 must benchmark ingest headroom, tensor throughput, supported operations, model memory, and training time before any hardware-sufficiency claim is made.

Cloud GPUs become useful for:

- Large hyperparameter sweeps.
- High-resolution diffusion experiments.
- Multiple large checkpoints.
- CUDA-only operations.
- Faster turnaround after the local design is stable.

The local machine is the proposed development environment. Cloud compute is an optional burst resource.

16.4 Cost envelope

Indicative prices as of July 10, 2026:

ITEM	EARLY RESEARCH COST	NOTES
Local model compute	$0 incremental cloud spend	Electricity and hardware depreciation still exist.
Public site	$0 personal/non-commercial	Vercel Hobby; commercial use should move to an appropriate plan.
Vercel Pro	$20/month	Current listed starting price.
Bookmap Digital+	$19/month	Data not included; not required for core research.
Cloudflare R2	$0.015/GB-month after free tier	Roughly $15/TB-month before operations; rights matter more than storage price.
Modal Starter	$0 plan fee with current included credit	Usage billed by second after credit.
Modal L4	about $0.80/GPU-hour	Derived from current per-second rate.
Modal A100 80GB	about $2.50/GPU-hour	A 20-hour run is roughly $50 before CPU/memory.
Modal H100	about $3.95/GPU-hour	Not justified until smaller models are exhausted.
Kraken API	no separate API fee stated	Data rights and account requirements remain separate.
Chainlink Data Streams	quote/subscription	Direct licensed use and retention scope must be negotiated.
Market-data rights	unknown	Potentially the largest non-compute cost.

These are marginal infrastructure estimates, not a project budget. Prices and entitlements change. Labor, legal review, data licensing, collection delay, security, monitoring, and partner integration are unpriced until Phase 0.

The listed platform prices and plan descriptions come from the providers' dated public pages 46–52, 60.

Current all-in budget: unknown. Current committed project team: not specified in this paper. Calendar estimate: not responsibly knowable until the data-rights path, pilot message rate, label prevalence, and local profiler are measured. Spending more on GPUs cannot compress the forward-data collection period or replace legal permission.

A credible pilot needs named ownership of at least four functions, even if one person covers several: data/replay engineering, model research, market-microstructure review, and data-licensing/legal review. Phase 0 must convert those functions into accountable people, a resourced schedule, and an all-in pilot budget before HEATWAKE asks a partner to commit engineering time.

The project can begin with local research and no AI credits. The unavoidable cost is time spent collecting trustworthy data. The potentially decisive cost is permission to retain, transform, train on, and commercialize market data.

16.5 Data scale

Storage is measured before architecture is fixed.

The recorder must report:

- Messages per second by feed and regime.
- Raw bytes and compressed bytes per hour.
- Snapshot frequency.
- Gap/reconnect rate.
- Derived tensor amplification.
- Retention tiers.

Raw events are preserved when rights permit. Derived frame caches can be regenerated and deleted. A selective corpus is created from immutable manifests; the final test is write-protected.

17. Bookmap: benchmark, interface, and possible partner

17.1 Do we need Bookmap?

No. A Bookmap-class liquidity surface can be rendered from licensed native order-book events. The heatmap is a deterministic visualization of resting size through time; the source data matters more than the brand of renderer.

Training on Bookmap screenshots is technically weaker and legally more complicated [46–47](), [61]():

- Screen capture loses event order and precision.
- UI state, crop, color, and zoom become spurious features.
- Raw additions and removals can be ambiguous.
- The visualization and underlying data may involve separate rights.
- Bookmap's standard Data Subscriber Services Agreement, including its use restrictions in Sections 1.2–1.4, does not supply the rights HEATWAKE would need for a commercial model trained on the data.

The correct sentence is:

> **HEATWAKE does not train on Bookmap screenshots. It applies licensed feed messages within documented provider semantics and renders its own semantic liquidity field.**

17.2 Why Bookmap could still matter

Bookmap can accelerate the project through:

- Deep expertise in order-flow visualization and data-provider semantics.
- Existing distribution to traders who understand the field.
- Historical/live workflows and replay.
- Add-on and overlay infrastructure.
- Access to expert readers for consensual annotation and evaluation.
- Potential data/licensing relationships.
- A credible environment in which Rosetta outputs can be overlaid on the heatmap.

The official Bookmap API supports modules and arbitrary screen-space overlays, creating a plausible path for a HEATWAKE add-on once the model is valid and permissions are explicit.

17.3 The proposition to a Bookmap founder

The proposition is:

> Bookmap made hidden market structure visible. HEATWAKE is an attempt to learn the latent dynamics of that structure, render the futures the model still considers possible, and show exactly when evidence is insufficient.

The evidence package should contain:

1. A live or recorded licensed event stream.
2. HEATWAKE's own deterministic field.
3. The latent future distribution.
4. Rosetta's visual and intervention-tested readout.
5. A timestamped shadow forecast.
6. Honest benchmark results against the raw book and the market.

Bookmap is a potential partner and deployment surface, not a hidden dependency.

17.4 A concrete first conversation

The first conversation is exploratory, not a request for capital or exclusivity. HEATWAKE would ask Bookmap to:

1. Challenge the event semantics, renderer, and failure model.
2. Identify a lawful path to non-display archival and model-training rights, if one exists.
3. Confirm the appropriate add-on/overlay integration boundary.
4. Help define an expert evaluation in which annotations and disagreements are retained.

In return, Bookmap would receive a low-risk way to test whether its expert users understand and trust probabilistic future-field overlays, influence an integration before its interface hardens, and evaluate a differentiated add-on hypothesis without committing its core platform. HEATWAKE would bear the research build; the collaboration would stop if the locked benchmark fails. If it passes, Bookmap would have the option to discuss a separately negotiated add-on or research collaboration. No ownership, exclusivity, data grant, or revenue split is implied by this paper.

If Bookmap does not participate, the core research can proceed only on a directly licensed venue feed with HEATWAKE's renderer. If Kraken, Chainlink, or the intended prediction-market source denies the required rights, the project switches to a source with explicit research/training permission or stops the affected branch.

18. Roadmap and gates

The first decisive experiment

Before building the complete product, run one narrow kill/no-kill test:

1. Secure a lawful private-research scope for one BTC venue.
2. Collect a pilot interval, measure dependence/event prevalence, and freeze a powered future test.
3. Render two fixed causal contexts and evaluate two fixed short horizons selected before the test.
4. Train three matched models: raw-message only, semantic-field only, and fused LWM-1A.
5. Score multistate transitions and price-path distributions on the untouched future using paired block inference.
6. Add the future-latent objective only as an ablation.

No Polymarket trading policy, narrative Rosetta, diffusion model, or quantum-inspired head is needed for this test. If the field or fusion provides no preregistered improvement in predictive performance, calibration, sample efficiency, or robustness, the visual-model thesis is stopped or reduced to an interface/retrieval product.

Phase 0 — Rights and protocol

Deliver:

- Written data-rights questions to Kraken, Chainlink, Polymarket, and any later provider.
- Frozen event schema.
- Renderer specification.
- Label definitions.
- Baselines, splits, metrics, and kill criteria.
- Threat model and jurisdiction policy.
- Named owners for data engineering, model research, microstructure review, and data/legal review.
- A resourced pilot schedule and all-in budget based on actual rights quotes and profiler measurements.

Exit gate:

- A lawful private research scope is clear.
- Commercial/public scope is either secured or explicitly deferred.

Phase 1 — Recorder and replay

Deliver:

- Provider-aware, depth-bounded Kraken L3/L2 and trade recorder.
- Chainlink reference recorder.
- Market discovery and official-outcome collector where permitted.
- Immutable Parquet event lake.
- Gap detection and clock telemetry.
- Bit-identical offline/live reconstruction tests.

Exit gate:

- The same tape reproduces the same book and tensor.
- Invalid intervals are automatically rejected.

Phase 2 — Field and simulated product

Deliver:

- Shared deterministic tensorizer.
- Multi-resolution semantic channels.
- Synthetic event generator isolated from real data.
- Zoomable interface matching the future model API.
- Permanent simulation labels.

Exit gate:

- The demo explains the actual planned system without implying a trained model.

Phase 3 — Baseline laboratory

Deliver:

- Engineered-feature models.
- Event-only model.
- Field-only model.
- DeepLOB-style baseline.
- Retrieval index.
- Chronological evaluation harness.

Exit gate:

- At least one learned representation beats honest trivial baselines on the calibration/test protocol.

Phase 4 — Latent dynamics

Deliver:

- Recurrent latent state.
- Future-latent distribution.
- Multistate transition and landmark-state heads.
- Multi-horizon path heads.
- Future-field decoder for inspection.

Exit gate:

- The latent future objective adds measurable value to at least one locked downstream task.

Phase 5 — Rosetta

Deliver:

- Visual future field.
- Probabilistic concept layer.
- Historical analogues.
- Counterfactual intervention panel.
- Constrained narrative gloss.
- Faithfulness evaluation.

Exit gate:

- Diagnostic outputs are calibrated, and sensitivity/model-internal intervention tests support their relationship to the forecast.

Phase 6 — Prediction-market shadow

Deliver:

- Contract discovery.
- Chainlink-aligned settlement state.
- Read-only/paper Polymarket mirror where permitted.
- Model-versus-market probability comparison.
- Timestamped prediction commitments.
- Execution-aware simulated policy.

Exit gate:

- Frozen live shadow performance remains within predeclared tolerances.

Phase 7 — Licensed partner deployment

Deliver:

- Written non-display/training/derived-data rights.
- A lawful venue or non-trading analytics scope.
- Optional Bookmap add-on.
- Operational risk controls.

Exit gate:

- Independent review approves claims, rights, and controls.

Phase 8 — Quantum-inspired research branch

Deliver only after the classical system:

- Tensor-network/Born-style density head.
- Parameter- and compute-matched classical baselines.
- Optional small offline quantum feature experiment.

Exit gate:

- Retain only on demonstrated predictive or efficiency value.

---

19. Governance, ethics, and security

19.1 No participant-identification claim

Kraken visible order IDs do not identify people. Polymarket's public data may expose wallet-linked activity, but trader profiling and real-time individual tailing are outside the core HEATWAKE thesis.

If public-wallet research is ever considered, it requires a separate purpose, legal review, privacy analysis, and manipulation/selection-bias safeguards. It is not included in or assumed by the liquidity model.

19.2 No intent claims from behavior alone

Order behavior may be described objectively:

- added,
- cancelled,
- replenished,
- executed,
- moved,

- persisted.

Words such as spoofing, manipulation, insider, or coordinated action require evidence beyond a heatmap pattern.

19.3 Operational controls

Before any execution capability:

- Read-only keys are the default.
- Withdrawal permission is never granted to model services.
- Trading credentials are isolated from training and the public site.
- A hard external kill switch exists.
- Position, loss, and rate limits exist outside the model.
- Stale or inconsistent feeds force abstention.
- Model and policy versions are immutable per decision.
- Every decision is logged before the outcome.
- No autonomous self-modification occurs in production.

19.4 Model security

The threat model includes:

- Poisoned or malformed market messages.
- Replay of stale events.
- Clock manipulation.
- Model/checkpoint substitution.
- Renderer-version drift.
- Unauthorized access to licensed raw data.
- Leakage of the final test set.
- Explanations that expose proprietary data.
- Adversarial attempts to trigger a known policy.

Raw data, checkpoints, and live credentials remain outside the public Vercel application.

---

20. Claims ledger

20.1 Supported by prior research

Prior work supports these limited statements:

- Limit-order-book dynamics contain short-horizon statistical information in several datasets.
- Order-flow information can be mapped into image-like channels for prediction.
- Temporal and spatial deep models can learn features from LOB data.
- Strong forecasting metrics may fail to produce actionable transactions.

- Structured-image generative models can produce future LOB states in research settings.
- Latent world models can predict and plan in compact representations in controlled domains.
- Continuous latent reasoning and non-text model communication are active research areas.
- Internal representations can sometimes be decoded or intervened on partially.

None of those statements proves HEATWAKE.

20.2 HEATWAKE hypotheses

The project will test whether:

- A semantic field's inductive bias improves out-of-time accuracy, calibration, sample efficiency, or robustness beyond event-only learning under matched budgets.
- Event/field fusion is more robust than either alone.
- A multimodal latent future objective improves downstream forecasts.
- Liquidity-structure transition hazards are calibrated and transferable.
- Rosetta can provide useful observability that survives sensitivity and model-internal intervention tests.
- The fused model improves paired proper scores and execution-aware decisions beyond a Polymarket-only quote baseline.
- Any advantage survives realistic execution and live drift.

20.3 Claims prohibited before evidence

Do not say:

- “HEATWAKE predicts the market.”
- “HEATWAKE knows the next movie.”
- “The model reads trader intent.”
- “The model identifies market makers.”
- “HEATWAKE is the first visual order-flow AI.”
- “The model thinks quantumly.”
- “Quantum superposition creates alpha.”
- “Rosetta translates every hidden thought.”
- “Backtest accuracy proves profit.”
- “Simulated outputs are forecasts.”

Whitepaper-safe language is:

> **HEATWAKE is a falsifiable research system for learning and visualizing conditional distributions over short-horizon liquidity futures.**

21. Conceptual origin

The project was motivated by a broader AI question: why serialize an internal representation into human language or rendered pixels before reusing it? Continuous latent reasoning and model-communication research illustrates that the question is real [14–22], but it is not evidence for market forecasting. HEATWAKE's technical foundation is a latent state-space forecaster over order-flow data.

The resulting pair is LWM-1 for conditional liquidity dynamics and Rosetta for separately tested observability. Its value depends entirely on ordered proof:

> **First reconstruct the market. Then learn its field. Then preserve its possible futures. Then decode those futures. Then calibrate them. Then compare them with the market. Then test execution. Only then discuss conviction.**

Conclusion

HEATWAKE attempts to turn visible order flow into auditable conditional forecasts—not to claim privileged access to hidden intent or a predetermined future. The Liquidity Wavefunction is a classical distribution over future trajectories. Rosetta is a separately tested observability surface. The policy is deterministic only after uncertainty, reliability, arrival-time cost, data quality, and legal eligibility are checked.

Success means the frozen system improves proper scores and execution-aware decisions on untouched future data. Failure means one or more preregistered hypotheses is rejected. Either result is more useful than a persuasive simulation mistaken for evidence.

Appendix A — Compact mathematical specification

A.1 History and rendering

Let (r_i) be the local monotonic time at which message (e_i) became available to the system. The realizable causal history is:

$$H_t^{avail} = \{e_i : r_i \le t\}.$$

For renderer version (\nu):

$$X_t^{(m)} = R_\nu^{(m)}(H_{t-W_m:t}^{avail}),$$

where (m) indexes temporal resolution. Exchange timestamps remain message attributes; they do not determine realizable inclusion.

A.2 Latent state

Let (E_t) be the event embedding, (X_t) the multiresolution field, and (C_t) causal context:

$$z_t = F_\theta(z_{t-1}, E_t, X_t, C_t).$$

A.3 Future distribution

For horizons (\mathcal{H}):

$$p_\theta(Z_{t+1:t+h} \mid z_t), \qquad h \in \mathcal{H}.$$

The field decoder is:

$$\hat{X}_{t+1:t+h} = D_\phi(Z_{t+1:t+h}),$$

but decision heads may operate directly on (z_t) and future latent samples.

A.4 Measurements

For task (j) and horizon (h):

$$\hat{y}_{j,h} = G_{j,h}(z_t, \mathcal{B}_t).$$

Measurements include structural transition hazards, price-path distributions, execution outcomes, and settlement probability.

A.5 Rosetta

Let (\mathcal{S}_t) be provenance, retrieval, and intervention state:

$$O_t = \mathcal{R}_\psi(z_t, \mathcal{B}_t, \mathcal{S}_t).$$

Rosetta output (O_t) is evaluated separately from task prediction.

A.6 Policy

The deterministic policy is:

$$a_t = \pi_\omega(\hat{y}_t, \text{quotes}_t, \text{costs}_t, \text{uncertainty}_t, \text{health}_t).$$

The action space is:

$$\mathcal{A} = \{\text{BUY\_UP}, \text{BUY\_DOWN}, \text{WAIT}, \text{ABSTAIN}\}.$$

Appendix B — Minimum experiment matrix

EXPERIMENT	INPUT	MODEL	OUTPUT	PASS CONDITION
E0 Integrity	Received message tape	Deterministic core	Depth-bounded rebuilt book	Provider checks, uncertainty masks, and tensor reproducibility
E1 Scalar baseline	Engineered causal features	Logistic/gradient-boosted model	Direction/hazards	Honest floor
E2 Event model	Raw event tokens	Causal temporal model	Same	Beats E1 on proper scores
E3 Field model	Semantic field	CNN/video model	Same	Tests visual inductive bias
E4 Fusion	Events + field	Dual encoder	Same	Beats E2 and E3 under matched budget
E5 Latent future	Events + field	Recurrent generative model	Future latents + tasks	Improves locked downstream metrics
E6 Rosetta	Frozen E5 state	Decoder/probes	Field/concepts/prose	Faithful interventions and calibration
E7 Market fusion	E5 + Chainlink + CLOB	Calibrated head	Settlement probability	Beats executable market baseline
E8 Shadow	Live read-only feeds	Frozen system	Committed forecasts	Meets predeclared live tolerances
E9 Amplitude	Frozen representation	Born/tensor head	Same probabilities	Beats matched classical head or removed

Appendix C — Data-rights checklist

Obtain written answers for every source:

1. May raw events be retained indefinitely?
2. May events be transformed into fields, features, labels, and embeddings?
3. May those derivatives train a private model?
4. May the model be used commercially?
5. May model weights be distributed?
6. May derived probabilities and signals be displayed publicly?
7. May delayed or sampled heatmaps be shown?
8. May raw or derived data be shared with cloud processors?
9. Is non-display usage separately licensed?
10. Are historical archives available?
11. What audit, attribution, deletion, and security duties apply?
12. Do rights survive termination?

Public API access is not treated as a substitute for these answers.

References and primary sources

Bracketed numbers in the text refer to this list. Platform facts and prices are dated snapshots and must be reverified before implementation or publication.

Market microstructure and LOB prediction

1. Zhang, Z., Zohren, S. & Roberts, S. DeepLOB: Deep Convolutional Neural Networks for Limit Order Books, 2018/2019. ↩
2. Sirignano, J. & Cont, R. Universal Features of Price Formation in Financial Markets, 2018.
3. Kolm, P., Turiel, J. & Westray, N. Deep Order Flow Imbalance, 2021/2023.
4. Lucchese, L., Pakkanen, M. & Veraart, A. The Short-Term Predictability of Returns in Order Book Markets, 2022.
5. Briola, A., Bartolucci, S. & Aste, T. Deep Limit Order Book Forecasting, 2024. The associated codebase is LOBFrame.
6. Lensky, A. & Hao, M. Learning to Predict Short-Term Volatility with Order Flow Image Representation, 2023.
7. Backhouse, A. et al. Painting the Market: Generative Diffusion Models for Financial Limit Order Book Simulation and Forecasting, 2025.
8. Nagy, P. et al. Generative AI for End-to-End Limit Order Book Modelling: A Token-Level Autoregressive Generative Model of Message Flow Using a Deep State Space Network, 2023.
9. Coletta, A. et al. Learning to Simulate Realistic Limit Order Book Markets from Data as a World Agent, 2022.
10. Nagy, P. et al. LOB-Bench: Benchmarking Generative AI for Finance—an Application to Limit Order Book Data, ICML 2025.
11. Cont, R., Kukanov, A. & Stoikov, S. The Price Impact of Order Book Events, 2010/2014. ↩
12. Huang, W., Lehalle, C.-A. & Rosenbaum, M. Simulating and Analyzing Order Book Data: The Queue-Reactive Model, 2013/2015.
13. Dubach, P. The Anatomy of a Decentralized Prediction Market: Microstructure Evidence from the Polymarket Order Book, 2026 preprint. ↩

Latent reasoning, communication, and world models

14. Hao, S. et al. Training Large Language Models to Reason in a Continuous Latent Space, 2024. ↩
15. Zhu, H. et al. Reasoning by Superposition: A Theoretical Perspective on Chain of Continuous Thought, 2025.
16. Zhang, Y. et al. Do Latent Tokens Think? A Causal and Adversarial Analysis, 2025 preprint.
17. Fu, T. et al. Cache-to-Cache: Direct Semantic Communication Between Large Language Models, ICLR 2026.
18. Liu, Y. et al. DroidSpeak: KV Cache Sharing Across Fine-Tuned Model Variants, NSDI 2026.
19. Hafner, D. et al. Learning Latent Dynamics for Planning from Pixels, 2018/2019.
20. Hafner, D. et al. Mastering Diverse Control Tasks through World Models, Nature 2025.
21. Bardes, A. et al. Revisiting Feature Prediction for Learning Visual Representations from Video, 2024.
22. Drozdov, K., Shwartz-Ziv, R. & LeCun, Y. Video Representation Learning with Joint-Embedding Predictive Architectures, 2024.

Interpretability, calibration, and research validity

23. Koh, P. W. et al. Concept Bottleneck Models, ICML 2020. ↩
24. Kim, E. et al. Probabilistic Concept Bottleneck Models, ICML 2023.
25. Adebayo, J. et al. Sanity Checks for Saliency Maps, 2018.
26. Jain, S. & Wallace, B. Attention Is Not Explanation, 2019.
27. Guo, C. et al. On Calibration of Modern Neural Networks, 2017. ↩
28. Geifman, Y. & El-Yaniv, R. Selective Classification for Deep Neural Networks, NeurIPS 2017.
29. Gneiting, T. & Raftery, A. Strictly Proper Scoring Rules, Prediction, and Estimation, 2007.
30. Bailey, D. et al. The Probability of Backtest Overfitting, 2016.

31. Bailey, D. & López de Prado, M. The Deflated Sharpe Ratio, 2014.

Quantum-inspired boundaries

32. Liu, J.-G. & Wang, L. Differentiable Learning of Quantum Circuit Born Machines, 2018.
33. Han, Z.-Y. et al. Unsupervised Generative Modeling Using Matrix Product States, 2017.
34. Huang, H.-Y. et al. Power of Data in Quantum Machine Learning, 2021.

Official platform and infrastructure sources

35. Kraken. Spot WebSocket v2 Level 3. ↩
36. Kraken. Spot WebSocket v2 Level 2 Book.
37. Kraken. Spot WebSocket v2 Trades.
38. Polymarket. Market Data Overview. ↩
39. Polymarket. CLOB Market WebSocket.
40. Polymarket. Real-Time Data Socket and Chainlink/Binance Price Streams.
41. Polymarket. Geographic Restrictions. ↩
42. Polymarket. BTC Up or Down 5-Minute Series. ↩
43. Polymarket. BTC Up or Down 15-Minute Series.
44. Chainlink. BTC/USD Data Stream.
45. Chainlink. Data Streams Documentation.
46. Bookmap. Layer 1 API Demo and Extension Documentation. ↩
47. Bookmap. Plans and Data Disclaimer.
48. Apple. Accelerated PyTorch Training on Mac. ↩
49. PyTorch. MPS Backend.
50. Modal. GPU Pricing.
51. Cloudflare. R2 Pricing.
52. Vercel. Plans and Pricing.
53. Kraken. Global Terms of Service. ↩
54. Kraken. Downloadable Historical Market Data.
55. Polymarket. International Terms of Use. ↩
56. Polymarket US. Trader Market Data. ↩
57. Polymarket US. FIX Market-by-Order Subscription.
58. U.S. Commodity Futures Trading Commission. QCX LLC d/b/a Polymarket US Designated Contract Market Filing.
59. Chainlink. Data Streams Data Sources.
60. Chainlink. Data Streams Billing. ↩
61. Bookmap. Bookmap Data Subscriber Services Agreement. ↩
62. Polymarket. CLOB Changelog. ↩
63. Polymarket. CLOB V2 Migration Guide.

Disclaimer

HEATWAKE is a research proposal. It is not investment advice, a solicitation, a performance claim, or an offer to trade. All current public-site forecasts are simulated unless explicitly replaced by timestamped reproducible results. No live execution should occur without legal eligibility, written data rights, operational controls, and independent review. Market trading can result in total loss.

HEATWAKE · V0.2 · JULY 10, 2026

Authored by William Keenan · K&E Studios Research

Verify this release