SYNTHEXLAB

ON-CHAIN SYNTHETIC ASSET RESEARCH

v0.1 RESEARCH DESIGN

Test the thesis. Before the market.

MODEL. REPLAY. STRESS. DRILL.

SynthexLab is a proposed on-chain synthetic asset research and risk sandbox. It brings an asset thesis, its data, its rules, and its failure conditions into the same experiment before live deployment is considered.

The research objective

Researchers would define simulated assets that track indices, commodities, interest rates, or thematic baskets, replay historical paths, apply adverse shocks, and rehearse oracle failures. The intended output is a reviewable evidence package: configuration, data provenance, assumptions, results, and unresolved questions.

Solana is the intended ecosystem. Early work prioritizes off-chain computation and reproducible experiment records. Any on-chain module or liquidity connection would require a separate implementation and review. A simulated asset carries no ownership, redemption right, or return entitlement in an underlying asset.

PROPOSED PRODUCT AND RESEARCH METHODS

No production deployment is claimed. Simulated assets have no live value. Backtests and drills do not establish future performance or system safety.

VERSION 0.1 / SEPTEMBER 2026 / ENGLISH

02 / SCOPE & ASSET LIFECYCLE

Define an experiment, not just an exposure.

A claim to track a market is not a complete specification. Each model needs explicit valuation, timing, rebalance, and exception rules.

Reference family	Required model boundaries
Indices	Historical constituents and weights, effective dates, and a clear distinction between price and total-return indices.
Commodities	Spot, futures, or a custom basket. Futures models must state contract selection, rolls, and term-structure assumptions.
Interest rates	Distinguish a rate level, a yield change, and the return on a rate-sensitive asset. Define accrual, tenor, calendar, and valuation.
Thematic baskets	Inclusion rules, weight caps, exits, and unavailable components, using membership actually knowable at each point in time.

The proposed lifecycle

Define > Freeze > Replay > Stress and drill > Review > Archive or iterate. A material model revision should create a new version while retaining earlier results and a change record. A model that fails review can remain useful for research, but it must not carry a production-ready label.

The definition records the research objective, reference assets, quote unit, starting benchmark, sampling frequency, rebalance calendar, fees, and slippage assumptions. Freezing adds the data manifest, configuration version, engine version, random seed, and run timestamp so another researcher can reconstruct the same experiment.

Deliver results as evidence

Reports should pair reference and simulated values with tracking deviations, drawdowns, cost attribution, scenario outcomes, incident timelines, and limitations. Parameter sensitivity and failed cases belong beside the baseline. A composite score must not hide distinct model risks.

> Initial boundary: virtual accounting and simulated positions. The proposed sandbox does not offer custody, redemption of real-world assets, or production trading services. Any expansion requires a separate product and security specification.

03 / PROPOSED ARCHITECTURE

Reproducible computation. Traceable publication.

The design separates research computation, provenance records, and on-chain state. Historical replay and production trading have different responsibilities.

Layer	Responsibility and output
Experiment workspace	Asset recipes, source selection, scenario editing, and result comparison; emits versioned configurations.
Data and provenance	Raw snapshots, licenses, unit and time normalization, and cleaning logs; emits an immutable data manifest.
Simulation engine	Deterministic event replay, rebalancing, costs, and fault injection; emits event logs and metrics.
Evidence and reporting	Configurations, source checksums, execution versions, and result summaries; supports independent recomputation.
Solana adaptation	Research into configuration registration, summary anchoring, permissions, and state. A later technical specification sets the scope.

The intended Solana implementation

Solana is the planned deployment target. The initial design performs replay off-chain and records necessary version identifiers, digests, and status on-chain. A digest can establish that files match; it cannot prove authentic inputs, a correct model, or unbiased execution. Independent recomputation and review remain necessary.

Later on-chain position or market modules need account and authority rules, fixed-point precision, boundary handling, replay protection, and idempotency. Experiments should include congestion, submission delays, update ordering, and unavailable services.

Trust boundaries

Researchers disclose configuration and result scope. Data credibility and licensing are assessed separately. Execution services expose versions and logs. Administrator powers must be visible. Material changes to sources, engines, or permissions should invalidate prior readiness reviews and trigger renewed validation.

> Initial acceptance target: the same data snapshot, configuration, and engine version reproduce the same event sequence and results. Any permitted numerical tolerance must be disclosed in advance.

04 / BACKTESTING / DATA & VALUATION

Backtesting begins with the data.

A meaningful replay acts only on information available at the time. It also preserves costs, missing observations, and failed outcomes.

Provenance and time alignment

Each dataset should identify its provider, acquisition time, coverage, frequency, time zone, trading calendar, quote unit, revision policy, and permitted use. Experiments retain both the time an observation occurred and the time a researcher could obtain it. When frequencies differ, the report must explain alignment, lags, imputation, and truncation.

Raw and transformed data remain separate. Every cleaning step should be logged, including corporate-action adjustments where relevant, removed records, unit conversions, and gap handling. A stored manifest should link each run to the exact input snapshot rather than a source URL whose contents can change.

An explicit sequence of events

The intended loop is: receive available data, value the model, evaluate rules, simulate execution, settle costs, and record state. Rebalancing can use only information available at that decision time. Execution latency, fees, slippage, and rebalance costs must be explicit. Financing or borrowing costs are separate inputs where relevant; an unknown cost must not silently become zero.

A valuation illustration

> For illustration only, a quantity-based basket can use:
> **V(t) = sum over i of [q(i,t) x P(i,t)] + C(t)**
> q is each component quantity; P is its valid reference price; C is cash net of cumulative costs.

Weighted indices, rate references, and futures strategies need their own methodologies. The illustration above is not a universal pricing formula. A method should specify when quantities change, how cash accrues, how stale inputs affect valuation, and which events can prevent a value from being published.

Make uncertainty visible

Reports should distinguish observed data from generated values. Synthetic history and imputed observations require prominent labels. If the required historical data is unavailable, the limitation should appear beside the result rather than being hidden in a general disclaimer.

05 / BACKTESTING / VALIDATION

Test the method, not only the outcome.

A strong historical curve can reflect bias or parameter selection. The research process should make those possibilities inspectable.

Potential bias	Proposed control
Look-ahead and revisions	Replay by availability time. Use historical data vintages and disclose later revisions.
Survivorship and selection	Retain exited constituents and failed experiments. Publish sample inclusion rules.
Overfitting and repeated search	Separate development, validation, and out-of-sample windows. Record trial counts and parameter search ranges.
Costs and missing values	Report multiple cost assumptions, flag gaps, and compare plausible treatments.

Validation across time and assumptions

The intended methodology includes rolling-window analysis and out-of-sample evaluation. Parameters should be fixed before a final holdout is evaluated. Results should show whether conclusions persist across adjacent parameter settings and data treatments, including less favorable choices.

A failed experiment remains part of the research record. If a model changes after a holdout is inspected, the report should identify that iteration and avoid presenting the same window as untouched validation. Limited coverage and small samples call for restrained claims about precision and robustness.

A consistent reporting contract

Each report should include maximum drawdown, tracking difference, tracking error, turnover, modeled costs, and the effective sample count. It must define metric calculations, observation frequency, and annualization conventions. Reference and simulated paths should use compatible units and time windows.

Cost attribution and sensitivity should explain why a result changes. Reports should preserve the configuration, input hashes, engine version, event log, random seed, and known limitations. A comparison is only meaningful when differences in method and data are visible.

> Historical backtests are descriptive experiments. They do not predict future returns or establish that a live asset will track a reference. Generated data must never be presented as observed historical performance.

06 / STRESS RESEARCH

Explore adverse market conditions.

Stress research seeks failure paths and sources of loss. A finite scenario set cannot prove that a system is safe.

Three ways to construct a scenario

Historical replay: revisit volatile periods or disruptions with their data limitations. **Hypothetical shocks:** explicitly change prices, volatility, correlation, costs, and latency. **Reverse stress:** define an unacceptable outcome and search for combinations that produce it.

Scenario family	Primary variables	What to observe
Price and gaps	Discrete shocks, sustained declines, opening gaps	Tracking deviations, rebalances, threshold responses
Correlation breaks	Changing dependence and rising concentration	Lost diversification and concentrated exposure
Liquidity contraction	Pool depth, trade size, fees, price impact	Execution-price deviation and executable size
Network and execution	Delay, failure, changed update order	Stale decisions, duplicates, recovery
Compound events	Market shock plus data or execution faults	Accumulated risks and interacting controls

An example, not a production threshold

A template could combine a 20% price decline, a 60% reduction in pool depth, and a 120-second update delay. These figures illustrate a scenario; they are not actual events, recommended thresholds, or completed results. Researchers should test a broader parameter grid and disclose coverage gaps.

Acceptance and attribution

Every run should predeclare its observation window, initial conditions, random seed, trigger thresholds, and acceptance criteria. Results should separate the effects of reference moves, trading costs, rebalance rules, and failure handling. Remaining operational is not a sufficient acceptance test.

> Proposed outputs include sensitivity maps, worst observed paths, incident timelines, recovery times, and uncovered conditions. Conclusions apply only to the disclosed models and scenarios.

07 / ORACLE FAILURE DRILLS

Treat bad data as a test case.

Ordinary price paths reveal only part of the risk. Stale data, outages, disagreement, and recovery need repeatable drills.

Injected fault	Proposed check and response research
Stale or missing inputs	Check age, heartbeat, and consecutive gaps; study restrictions on new exposure and transition to a pause.
Source disagreement or jumps	Compare deviations and confidence information; study isolation, reduced capability, and review.
Out-of-order or duplicate updates	Validate timestamps, sequence, and idempotency; prevent older values from overwriting confirmed newer values.
Wrong unit, precision, or sign	Apply range and format checks; malformed inputs must not silently enter valuation.
Bursts after an outage	Do not mistake delayed updates for current information; study replay, reconciliation, and staged recovery.

A proposed state machine

Normal > Degraded > Paused > Recovery observation > Normal. Published rules and permissions govern transitions. Degraded data is not trustworthy by default. Insufficient valid inputs should retain the fault flag and restrict activity, never produce an invented price.

Fallback sources need assessment of quality, independence, and suitability. Providers may share an upstream feed, so source count alone does not establish resilience. Log every switch, trigger, price, and responsible authority.

Recovery needs fresh evidence

Recovery should require consecutive valid observations, renewed agreement, cleared backlogs, and reconciled state. Report valuation jumps, accumulated errors, recovery time, and manual interventions. Automatic and manual recovery both need protection against unauthorized changes and duplicate execution.

> These controls remain proposed and require implementation and validation. A drill must not be described as a deployed contract safety guarantee or a certification of an oracle provider.

08 / MARKET STRUCTURE & DELIVERY

Connect research to market structure.

The intended Solana path brings initial liquidity, price discovery, and execution conditions into the research agenda.

Liquidity pools as experimental objects

Pool research would specify the quote asset, depth, fees, trade-size distribution, and withdrawal paths. Pool structures require distinct models, with pricing rules used only where their assumptions apply. Reference value and execution price must be separate so illiquidity is not mistaken for a reference change.

The ecosystem direction explores an **X-marked community-led pool path** and its potential role in price discovery. The pool, pairing, and connection timing require formal disclosure. This is a design direction, not a verified partnership, deployment, endorsement, or venue relationship. Any connection requires disclosure of its mechanism, dependencies, and risks.

Progress through evidence gates

Stage	Deliverable	Gate to progress
Research specification	Data model, methodology, fault dictionary	A third party can understand and reconstruct the method
Sandbox prototype	Virtual assets, replay, scenarios, reports	Repeatable runs with visible bias and limitations
Chain adaptation	Solana prototype and authority model	Boundary, failure, and security tests completed
Independent readiness review	Technical review, operating plans, disclosures	Findings resolved and a justified deployment decision

Token and participation policy

This version sets no token supply, allocation, yield, governance rights, or contract address. A future token requires separate disclosure of purpose, dependencies, allocation, restrictions, and holder rights, distinct from simulated assets. Research participation promises no token or economic return.

> The roadmap uses conditional gates rather than unconfirmed launch dates. Evidence, implementation quality, and independent review determine whether the next stage is justified.

09 / READINESS & LIMITATIONS

Readiness requires independent evidence.

Publishing uncertainty is part of the research result. A successful simulation cannot substitute for a review of a real system.

Minimum review before a live environment

Review area	Evidence to obtain
Data	Source permissions, update behavior, historical vintages, and documented exception handling.
Model	Independent recomputation, parameter sensitivity, out-of-sample evaluation, and reverse stress tests.
Implementation	Review of authorities, numerical precision, state transitions, dependencies, and recovery behavior.
Market	Assessment of actual depth, manipulation surfaces, withdrawals, and reference-price deviations.
Operations	Monitoring, incident responsibilities, upgrades, pause authority, and recovery procedures.

Material limitations

Models can omit behavior. Data may be incomplete, delayed, revised, or unrepresentative. Stress can change liquidity and participant reactions. Code, services, signatures, permissions, and procedures can fail. Live use also requires independent legal and compliance assessment.

Scenarios cannot cover all regimes, adversarial actions, or combined failures. Oracle and pool models may miss live behavior. Parameters and data treatment can change even a reproducible result.

What passing a test means

A pass means a specified model met specified criteria under specified conditions. It cannot establish future performance, reliable tracking, redeemability, or fund safety. A pass is not ongoing validation. Reports must flag important assumptions that remain unverified.

> Readiness is version-specific. Changes to models, data, code, privileges, dependencies, or markets can require new evidence.

10 / TERMS & PROJECT STATUS

Make the boundaries as clear as the thesis.

SynthexLab aims to make research inspectable: what was assumed, what was tested, what failed, and what remains unresolved.

Terms used in this paper

Term	Meaning in this research design
Simulated asset	A research object calculated under disclosed rules. It has no live value or rights in an underlying asset.
Evidence package	Configuration, provenance, versions, logs, results, and limitations grouped for review and recomputation.
Oracle failure drill	An experiment that perturbs reference inputs and observes the model and control response.
Production readiness	A status requiring a separate review. Neither this paper nor a backtest curve grants it.

Project status and scope

This paper describes the SynthexLab concept. It claims no operating performance, audited contract, or confirmed commercial relationship. Website graphics and results are demonstrations unless accompanied by reproducible data and an experiment description.

The architecture, lifecycle, controls, metrics, and delivery gates are design intentions and may change. Future releases should distinguish implemented features from hypothetical ones and identify the evidence behind each claim.

Version and use

v0.1 Research Design
September 2026 / English edition

This document is for research discussion. It is not an asset offering, investment recommendation, or promise of returns. Official, confirmed disclosures should define any eventual launch schedule, feature scope, token arrangement, and deployment status.

> Define the assumptions. Replay the history. Rehearse the failures. Then consider deployment.