PRAXIFI

A Non-Custodial Financial Policy Infrastructure for Digital Wealth

From Portfolio Diagnostics to Owner-Authorized Financial Policies

Version: 1.3 - Public Release

Publication: August 2026

Authors: Mohammad Saee Ghaemi | Alireza Heidari

AI Use Disclosure: Artificial intelligence tools were used solely to support drafting, language refinement, and editorial review of this whitepaper. Praxifi's core concept, product architecture, system logic, policy design, and strategic direction were developed by the Praxifi team.

Status: Public Release. The Praxifi web application is live, and Base Mainnet contract addresses are publicly listed in Praxifi documentation for the current policy path. Capability maturity is stated separately for each component. This release does not claim independently audited Production operation, verified real-funds execution, regulatory approval, global compliance, or independent scientific validation of Portfolio Healthiness.

Publication Status

This document is the public release of Praxifi Whitepaper V1.3. It describes Praxifi’s research foundations, live application status, intended architecture, product boundaries, and current development status. Public release of the document is separate from Product or Mainnet operational readiness. It does not represent an independent smart-contract audit, regulatory approval, legal inheritance determination, guaranteed execution service, or independently validated portfolio model.

Evidence and Implementation Status

The table below separates current application status, internally reported implementation evidence, deployment evidence, and independently validated evidence. Public availability of the web application or a Mainnet contract address does not by itself establish audited Production operation.

Area	Current status	Evidence status
Praxifi Web Application	Live	Founder-confirmed public deployment; current implemented web components are functioning. This does not establish on-chain Production readiness.

Area	Current status	Evidence status
Wallet Authentication	Operational	MetaMask connection, nonce issuance, wallet-signature verification, protected session, and dashboard access are implemented in the deployed application; independent penetration testing is not completed.
Portfolio Healthiness	Operational in the live application; implemented and internally tested	HS-V1 engine/API/provider/UI implementation is internally evidenced. Independent methodological validation, peer review, and external calibration remain pending.
Scheduled Transfer	Mainnet deployed on Base; application configuration and review available	Public contract-address documentation and internal deployment/test records exist. Real-funds operational execution and independent audit are not independently verified. Current automation eligibility is limited to USDC and USDT.
Basic Inheritance	Mainnet deployed on Base; application flow implemented	Public contract-address documentation and internal deployment/test records exist. Legal inheritance is not determined by the Product. Real-funds operational execution and independent audit are not independently verified.
Refundable Inheritance	Mainnet deployed on Base; application and contract lifecycle implemented and internally tested	Application/runtime/E2E evidence is internally reported. Independent audit, launch-jurisdiction legal review, and real-funds operational verification remain pending.
Worker and Poller	PARTIAL	Dry-run, simulation, reconciliation, runtime E2E, production-health, and monitoring artifacts are internally reported. Sustained Production operation, failover, SLOs, and real-funds runtime are not independently verified.
Smart contracts	Mainnet deployed on Base	Praxifi public documentation lists Base Mainnet contract addresses for Basic Inheritance, Scheduled Transfer, and Refundable Escrow. Independent audit is not completed; Polygon policy deployment is not established in this release.
Fee settlement	Implemented and internally tested; Production operational verification pending	Fee logic and internal tests are reported. Real-funds settlement, final fee-recipient governance, and Production economic edge cases are not independently verified.
Notifications	Implemented at application level; delivery verification pending	Notification scheduling/templates and lifecycle hooks are internally reported.

Area	Current status	Evidence status
		External delivery reliability, fallback, and regional availability are not independently verified.
Mainnet operation	Base Mainnet contract deployment confirmed internally; Mainnet operational status not claimed	Contract deployment is distinct from verified real-funds operation, audited Production launch, and Polygon Mainnet deployment.

Current Application Status

The Praxifi web application is live. The deployed application currently supports MetaMask wallet interaction and wallet-signature authentication, protected dashboard access, Portfolio Healthiness diagnostics, policy configuration and review interfaces, and status/history views for implemented Product components. These application capabilities are described in present tense where current implementation evidence supports them. The current public policy path is Base Mainnet. USDC and USDT are currently automation-eligible; WETH may appear in supported portfolio diagnostics but is not currently automation-eligible.

Application deployment and blockchain deployment are separate states. Praxifi public documentation lists Base Mainnet contract addresses for the three Version 1 policy contracts, while sustained real-funds operational use, independent smart-contract audit, complete Production operations, and Polygon policy deployment remain separately qualified. Website availability does not establish regulatory approval or independent scientific validation.

Document Status and Reading Guide

Praxifi is a live web application and an evolving non-custodial financial-policy infrastructure for digital assets. This whitepaper describes the research thesis, Version 1 product scope, current application behavior, technical architecture, security model, economics, legal uncertainty, roadmap, deployment evidence, and remaining evidence boundaries.

“Non-custodial” describes the current wallet-based architecture in which Praxifi does not receive the user’s private key or general authority over the wallet. It is not a universal legal classification, does not eliminate operational dependencies, and requires separate qualification for contract-controlled or escrow-like lifecycle states.

The document uses the following status labels consistently:

Status	Meaning
Live	Publicly deployed and accessible at application level. This does not imply Mainnet or Production readiness.
Operational	The relevant application-level function is working in the current deployed application.
Implemented and internally tested	The project reports working implementation and internal tests. Independent reproduction may still be unavailable.
End-to-end validated	The complete defined lifecycle has been exercised within the stated test environment. The environment must be stated.
Local-chain validated	A flow has been exercised against local contracts or a local blockchain environment. This is not Mainnet evidence.
Architecture validated	The intended components, permissions, states, and controls have been designed and reviewed, but operation may not be demonstrated.

Status	Meaning
Partially implemented	Material parts exist, but the complete runtime, monitoring, failure handling, or release evidence remains incomplete.
Production verification pending	Production monitoring, security, governance, operational approval, or real-funds evidence remains incomplete.
Mainnet deployed	A specified contract has been deployed to the stated Mainnet. This does not by itself establish safe or sustained Production operation.
Mainnet operational	The deployed system is processing intended Mainnet transactions under defined operational controls. This status is not claimed unless supported by evidence.
Independent audit completed	An external auditor has reviewed the exact stated implementation scope. This status is not claimed in this release.
Independent validation pending	The capability may function, but external methodological or technical validation has not been completed.
Research-stage	The capability or methodology remains an active research question.
Future version	The capability is not part of the current Product offering.
Blocked dependency	Progress depends on a specific external component, deployment, version, or audit that has not been verified.
Not currently supported	The capability is outside the current supported Product path.
Removed	The concept is no longer part of the current Product direction.
Unable to independently verify	Internal evidence or founder confirmation may exist, but independent reproduction or external verification is unavailable.

This whitepaper publicly describes Praxifi's technical design, current Product and deployment status, known legal uncertainties, and areas of potential regulatory exposure. It does not determine Praxifi's legal classification, establish compliance in any jurisdiction, or replace jurisdiction-specific legal, regulatory, or tax advice.

Naming and Scope

The official product name is **Praxifi**. Praxifi is designed to help users understand selected characteristics of a self-custodial portfolio and create narrowly authorized digital-asset policies. It does not receive the user's private key, reconstruct lost keys, determine lawful heirs, manage investments, promise portfolio improvement, or guarantee execution or returns.

In this whitepaper, "Inheritance" is a product-category term for an inactivity-triggered technical transfer policy. It does not determine legal inheritance, identify lawful heirs, establish succession rights, or make the designated recipient a legal beneficiary.

PraxiHub is a distinct research and knowledge platform that informs, examines, and may critically evaluate assumptions underlying digital-finance products, including Praxifi. PraxiHub is not a regulator, legal authority, investment adviser, or automatic independent verifier of Praxifi. Its credibility depends on disclosed editorial governance, methodology, funding relationships, conflicts of interest, correction practices, and publication versioning.

Abstract

Digital assets can remain recorded on-chain even when the person expected to use them can no longer access a wallet, monitor a portfolio, or initiate the next transaction. This is a problem of practical access, operational continuity, and execution of intent. It is not, by itself, a determination of legal ownership, death, incapacity, or succession.

Praxifi now operates a live web application that combines an experimental portfolio diagnostic with owner-authorized policy interfaces. The components still differ in maturity. HS-V1 is operational in the live application and remains independently unvalidated. Scheduled Transfer and Basic Inheritance are implemented in the application and have Base Mainnet contract deployments recorded by the project. Refundable Inheritance has application/runtime implementation, internal end-to-end evidence, and a Base Mainnet escrow deployment recorded by the project. Worker and Poller status remains PARTIAL because sustained Production operation, failover, service-level performance, and real-funds runtime have not been independently verified. No independent smart-contract audit or verified Mainnet Production launch is claimed.

For early clarity: “Inheritance” is a qualified Product label for an inactivity-triggered technical transfer, and “non-custodial” describes the intended wallet-based design rather than a universal legal classification. Contract-controlled or escrow-like Refundable states require separate analysis.

The owner selects the network, token, amount, designated recipient, condition, and cancellation rights. A technical transaction executor may submit an eligible transaction, while contract logic is designed to restrict execution to the authorized policy parameters. Base Mainnet contract addresses are publicly listed in Praxifi documentation and disclosed in Appendix C. Independent audit results, sustained real-funds operational evidence, and a Polygon policy deployment are not established by this release.

The broader thesis is that digital wealth may increasingly be managed through understandable, narrow, and revocable policies rather than disconnected manual transactions. Analysis may assist a decision; authority must remain with the user; and execution must depend on explicit authorization and verifiable conditions. Artificial intelligence may help explain data or draft a policy, but it must not obtain unrestricted authority over capital.

Contents

PART I - Origin and Strategic Thesis

1. Introduction

Blockchain systems allow a person to authorize transactions directly through cryptographic credentials. That technical control does not remove the need for continued access, attention, judgment, and operational capacity. A wallet may still show a balance when the person associated with it cannot sign, cannot recover access, or is no longer able to carry out a previously intended action [1,2,11].

Praxifi connects two needs that are often handled separately. The first is diagnosis: helping a user understand selected concentration, liquidity, asset, protocol, stablecoin, and operational characteristics of a supported portfolio. The second is policy execution: allowing the user to define a narrow instruction that may remain eligible over time without transferring unrestricted control to the platform.

The owner remains the source of authority. A diagnostic result can inform a decision, but it cannot move assets. A policy can move assets only after the owner reviews and authorizes its exact scope and the relevant technical condition is satisfied.

2. Why Praxifi Was Created

Praxifi grew out of research into wealth management, portfolio construction, behavioral finance, robo-advisory, digital assets, and blockchain automation. That work suggested a recurring distinction between deciding and executing. A financial plan may be reasonable and still fail because the person responsible for it delays, reacts under stress, forgets an obligation, loses access, or becomes unavailable [3-6].

The project therefore moved from the question, “How can a better portfolio be designed?” toward a more operational question: “How can a selected financial decision become a persistent, understandable, and limited policy?” A personal experience during a flight made the continuity problem concrete for the founders: a system that depends on one person to perform every future action remains fragile even when its assets are technically secure.

This origin explains why Praxifi begins with diagnosis, continuity, and scheduled execution rather than predictive trading. These are narrower problems that can be described through explicit inputs, observable conditions, cancellation rights, and evidence.

3. The Meaning of PRAXI

PRAXI represents the conversion of financial intent into a structured policy. A policy should identify:

- the person or account granting authority;
- the network and token;
- the gross amount and intended net amount;
- the designated recipient;
- the condition that permits execution;
- the cancellation, renewal, or expiry rules;
- the fee version accepted by the owner;
- and the evidence needed to verify creation, status, and outcome.

A transaction answers what should happen now. A policy answers what may happen if a defined condition becomes true.

4. Core Problems

Practical access can fail

Blockchain records can persist while the ability to use the associated assets is lost or interrupted. Legal entitlement, technical control, practical access, and ability to dispose of an asset are related but distinct concepts [11].

Self-custody does not provide continuity automatically

Self-custody reduces dependence on a centralized custodian, but it does not by itself provide a safe process for prolonged inactivity, loss of access, death, incapacity, institutional dissolution, or an unavailable signer.

Financial intent is usually temporary

Most wallet actions require a new signature at the time of execution. Long-term intent is rarely represented as a constrained policy with a lifecycle, cancellation state, and evidence trail.

Automation is fragmented

Wallets, transfer schedulers, analytics, notification systems, smart-account modules, estate-planning tools, and execution services often use different control models. A user may not be able to see how these components interact or which one can move assets.

Permissions may exceed the intended action

ERC-20 allowances can remain active beyond the immediate interaction. Long-lived policies therefore require amount limits, destination limits, chain binding, cancellation, monitoring, and token-specific validation [23,26].

Execution is vulnerable to human and operational inconsistency

Predefined rules may reduce selected delays or behavioral deviations, but they do not make an underlying strategy correct and do not guarantee superior financial outcomes [4,5].

5. Product Thesis

Praxifi’s Version 1 thesis is deliberately narrow: portfolio diagnostics should explain selected portfolio conditions, while owner-authorized policies should carry out a limited set of future actions. The Portfolio Healthiness layer is diagnostic and experimental. Basic Inheritance and Scheduled Transfer test policy definition, authorization, monitoring, cancellation, execution, reconciliation, notification, and evidence without depending on market prediction.

The longer-term thesis is that the same coordination model may support other risk, transfer, liquidity, and allocation policies. Every expansion must remain gated by evidence, technical security, operational readiness, legal analysis, and understandable user consent.

PART II - Research Foundation and PraxiHub

6. The Role of PraxiHub

PraxiHub is a research and knowledge platform focused on blockchain, digital assets, WealthTech, programmable finance, AI in wealth management, portfolio risk, and financial automation. It is the scientific and intellectual foundation informing Praxifi, but it should not function as a marketing department or as automatic proof that Praxifi works.

PraxiHub may support Praxifi in four ways:

1. define and refine the problems being addressed;
2. review academic, technical, institutional, and legal evidence;
3. translate findings into testable product hypotheses;
4. publish limitations and findings that may contradict the product narrative.

Operational meaning of independence

PraxiHub's independence should be assessed through process rather than branding. A credible operating model should disclose:

- editorial decision rights;
- funding and founder relationships;
- conflicts of interest;
- source-selection and review methods;
- whether a publication was internally or externally reviewed;
- correction and retraction procedures;
- version history;
- and separation between a research conclusion and a product decision.

PraxiHub now publicly presents research-ethics, correction, and relationship/conflict disclosures. These controls improve transparency but do not make PraxiHub an independent auditor, certification body, regulator, legal authority, or investment adviser. Founder, funding, editorial, and review relationships should continue to be disclosed at publication level.

Evidence standard

A source can motivate a design choice without proving the effectiveness of a particular implementation. PraxiHub and Praxifi should distinguish:

- external research evidence;
- primary standards and official documentation;
- internal project reports;
- public code and tests;
- deployment evidence;
- independent audit evidence;
- founder-approved product decisions;
- and unresolved assumptions.

Research Pillar I - Digital Asset Continuity

7. Digital Asset Continuity

Digital Asset Continuity studies how practical access, transaction capability, and owner-authorized instructions can remain operational across disruption. It includes but is not limited to inheritance-related use cases.

In self-custodial systems, the blockchain can verify signatures and state transitions, but it does not independently know whether a person is alive, legally capable, unavailable, coerced, or permanently unable to access a key. An inactivity signal proves only that an expected signal was not received within a defined period.

Technical control and legal entitlement

Blockchain control, practical access, and legal entitlement are distinct concepts. A private key can enable transaction authorization, but possession or use of that key does not necessarily determine lawful title. Loss of a key may prevent practical disposition without extinguishing legal rights. An on-chain transfer may change technical control without resolving claims under applicable law [11].

Praxifi therefore uses **Continuous Ownership** as a defined design principle, not a legal conclusion:

> Continuous Ownership is the continuity of owner-authorized control instructions across periods when the owner may be unable to act directly. It does not independently establish, determine, or transfer legal title.

Beyond inheritance

Continuity planning may be relevant to:

- prolonged inactivity;
- temporary or permanent incapacity;
- loss of authentication credentials;
- an unavailable institutional signer;
- inaccessible multisignature participants;
- emergency transfers;
- delayed distributions;
- or a predefined future transfer.

Praxifi does not decide who should receive assets. The owner defines a designated recipient and the technical conditions of the policy. Whether that recipient is legally entitled to retain the asset is a separate question.

Inactivity as a programmable condition

A user may define a period during which a valid activity confirmation is expected. A valid confirmation can renew the deadline. If no valid confirmation is received, the policy may become technically eligible for execution.

Conservative design requires:

- a sufficiently long owner-selected period;
- repeated reminders;
- simple renewal;
- clear cancellation rights;
- explicit network, token, amount, and recipient scope;
- balance and allowance readiness checks;
- transparent evidence;
- and an execution delay or other safeguard where the product design requires it.

Inactivity must never be presented as proof of death, incapacity, abandonment, identity, or legal competence.

Research questions

- How should false eligibility be reduced without creating intrusive identity surveillance?
- How should long-lived approvals expire, renew, or migrate?
- How should continuity interact with legal succession, tax, sanctions, and disputed entitlement?
- How can the system remain usable if a network, token, executor, RPC provider, or interface changes?
- How should active policies be migrated if a contract is replaced?
- Which events require an automatic block, an owner warning, or manual review?

The objective is not to claim that continuity is solved. It is to design a limited technical mechanism whose authority, evidence, and residual risk remain visible.

Research Pillar II - Autonomous Wealth

8. Autonomous Wealth

Praxifi uses **Autonomous Wealth** to describe a model in which owner-defined financial policies can remain active without constant manual supervision. Autonomy does not mean that an algorithm owns, manages, or freely reallocates capital. It means that execution can occur inside an explicit authorization boundary.

A policy contains four elements:

- **Intent:** the outcome the user wishes to achieve;
- **Condition:** the observable state that permits action;
- **Constraint:** the token, amount, recipient, network, timing, frequency, and cost limits;
- **Execution:** the deterministic state transition and evidence produced.

Why automation begins with deterministic actions

A deterministic policy should produce the same permitted action when the same verified conditions are present. Scheduled transfers and inactivity-triggered policies are suitable early cases because they do not require the platform to predict markets or select investments.

Predefined rules may reduce execution inconsistency, missed deadlines, and selected behavioral deviations. They do not guarantee better returns, lower total risk, or a suitable financial strategy [4,5].

Layers of a future policy system

- **Protection:** continuity, cancellation, recovery research, and emergency controls;
- **Management:** scheduled or recurring transfers and administrative instructions;
- **Risk:** monitoring concentration, liquidity, collateral, or protocol exposure;
- **Allocation:** owner-authorized rebalancing or capital deployment within a mandate;
- **Optimization research:** testing whether capital can be used more efficiently without obscuring risk.

Version 1 begins with diagnosis, Protection, and Management. Risk and allocation execution remain later-stage research and development.

AI boundary

AI may assist with explanation, data classification, scenario analysis, detection of missing constraints, or conversion of natural-language intent into a draft policy. The architecture must keep four stages separate:

1. analysis;
2. recommendation or draft;
3. owner review and authorization;
4. deterministic execution.

AI-generated output must not silently expand permissions, alter a designated recipient, increase an amount, or obtain independent authority over capital. Explainability and human review are important in financial contexts, but explainable output does not establish correctness [37,38].

Research Pillar III - Rule-Based Finance

9. Rule-Based Finance

Rule-Based Finance is the representation of selected financial decisions through predefined, constrained, and verifiable policies. It separates prediction from response.

A prediction asks what may happen. A policy asks what the system is permitted to do when a defined condition is observed. Forecasts and analysis may inform policy design, but the executable rule should remain explicit and reviewable.

Minimum policy definition

A complete rule should identify:

1. the authorizing account;
2. the network and asset;
3. the maximum gross amount;
4. the designated recipient;
5. the eligibility condition;
6. the cancellation, renewal, and expiry rules;
7. the accepted fee version;
8. the caller permitted to submit execution;
9. and the evidence needed to verify the outcome.

Policy engine layers

- **Interface:** explains the action, condition, cost, and remaining risk;
- **Authorization:** verifies the wallet, signature, approval, nonce, chain, and session;
- **Condition:** evaluates time, inactivity, status, balance, allowance, or approved external evidence;
- **Execution:** submits only the permitted call;
- **Evidence:** records policy hashes, signatures, states, events, transaction hashes, and reconciliation results;
- **Coordination:** a future layer for priorities, asset reservations, conflicts, and portfolio-wide limits.

Failure-aware design

A policy is incomplete unless it describes failure. Relevant states include insufficient balance, revoked allowance, unsupported token behavior, paused or blacklisted stablecoin, stale data, executor outage, RPC inconsistency, excessive gas, contract pause, chain reorganization, duplicate submission, and service shutdown.

Failure should be represented as an explicit state, not silently treated as successful execution.

Research Pillar IV - Resilient Financial Infrastructure

10. Resilient Financial Infrastructure

Financial automation has value only if it continues to behave safely under compromise, unavailability, revoked authorization, network disruption, and long time horizons. Resilience does not mean that failure can be eliminated. It means that authority is limited, failure is observable, and recovery paths are defined.

Least privilege

Each policy should receive only the authority required for the selected token, amount, recipient, network, condition, and execution window. Material parameters should not be editable by the technical transaction executor.

Long-lived permission risk

Policies may remain active for months or years. During that period, token contracts, wallet standards, RPC infrastructure, fee conditions, executor credentials, and legal requirements may change. Long-lived authorization therefore requires monitoring, expiry or migration rules, and user-visible status.

Executor resilience

The executor is intended to deliver eligible transactions, not define financial intent. Its key should be replaceable through a controlled process, but replacement authority creates its own governance risk. Secure design may involve managed signing, hardware-backed keys, rate limits, allowlists, multisignature administration, transparent events, monitoring, and a tested revoke path.

Frequent credential rotation is useful only when each credential has limited scope, an explicit lifetime, protected issuance, and immediate revocation. Rotation alone does not make an executor secure.

Evidence of intent

Policy integrity may be supported through structured owner signatures, chain-bound domains, nonces, deadlines, canonical policy hashes, on-chain events, transaction receipts, and application-to-chain reconciliation [24,25]. Any material policy change should require a new owner authorization and a new policy hash.

Research Pillar V - Programmable Portfolio Intelligence

11. Programmable Portfolio Intelligence

Programmable Portfolio Intelligence examines how wallet and market data can be converted into understandable diagnostics, evidence, and owner-reviewed policy drafts. It is the analytical layer between raw data and any authorized action.

Continuous monitoring does not imply continuous trading. A system may identify a concentration, unsupported asset, liquidity limitation, or protocol exposure without moving funds. Execution should require a separate policy and authorization.

Portfolio Healthiness as a research hypothesis

The Portfolio Healthiness Score is a versioned, diagnostic, experimental, and heuristic model. It is not:

- investment advice;
- a suitability assessment;
- a credit rating;
- a prediction of returns;
- an objective universal standard;
- or a guarantee that a portfolio is safe or well designed.

The model has not yet received independent validation, peer review, large-sample empirical calibration, prospective outcome testing, regulatory suitability validation, or complete cross-market validation. Its weights and thresholds are product calibration parameters, not scientific facts.

The Version 1 objective is narrower: disclose the data used, show which factors affected the result, identify missing evidence, and help the user decide what deserves review.

Research Pillar VI - Capital Allocation Research

12. Capital Allocation Research

Earlier product language used the label **Excess Return** for a future layer. That phrase can imply promised outperformance, alpha generation, investment management, or a managed financial product. V1.3 therefore uses **Capital Allocation Research** as the more accurate public label while preserving the founder's long-term interest in risk-aware capital efficiency.

This pillar studies how an owner might define a mandate containing permitted assets, allocation bands, liquidity reserves, protocol limits, transaction-cost constraints, suspension conditions, and revocation rights. It does not describe a current product offering.

Potential research areas include threshold rebalancing, risk budgets, recurring allocation, liquidity management, and multi-protocol exposure. These capabilities introduce oracle, market, liquidity, smart-contract, governance, tax, regulatory, and suitability risks. Praxifi does not currently claim to provide autonomous investment management, predictive trading, guaranteed yield, or guaranteed excess return.

PART III - Version 1 Product Model and User Experience

13. Product Overview and Version 1 Audience

Version 1 is a live wallet-based diagnostic and policy application for supported self-custodial users. The Product names Basic Inheritance and Refundable Inheritance describe qualified technical continuity flows, not legal succession services. The non-custodial design applies to wallet-based phases; any contract-controlled or escrow-like state is described separately.

The current Version 1 audience is primarily self-custodial digital-asset users who:

- use MetaMask;
- hold supported assets on the current Base Mainnet public path;
- want an informational portfolio diagnostic;
- want to schedule a future token transfer;
- or want to create an inactivity-triggered technical continuity policy.

Version 1 is not presented as institution-ready. Institutional templates, multi-party authorization, policy conflicts, service-level commitments, compliance integrations, and advanced governance remain later-stage requirements.

14. Product Layers

Analyze

The live Analyze layer presents asset allocation and the experimental Portfolio Healthiness diagnostic. It does not move assets or issue a personalized investment recommendation.

Protect

The Protect layer provides the current Basic Inheritance configuration and review path for technical continuity policies. Basic Inheritance refers to an inactivity-triggered transfer to a designated recipient. In this whitepaper, “inheritance” is a product-category description. It does not determine legal inheritance, identify lawful heirs, prove death or incapacity, or establish succession rights.

Manage

The Manage layer provides the current Scheduled Transfer configuration and review path: a future-dated, owner-authorized transfer that remains cancellable before execution, subject to the policy lifecycle and contract state.

Capital Allocation Research

This is a future research layer. It is not enabled in Version 1 and does not promise outperformance, managed allocation, or investment returns.

15. Current Scope

Area	Version 1 scope and boundary
Web application	Live. The current deployed application exposes implemented dashboard, authentication, diagnostics, policy configuration/review, status, and supporting information interfaces.
Networks	Base Mainnet is the current public policy chain, with contract addresses listed in Praxifi documentation. Polygon PoS, chain ID 137, remains within approved Version 1 scope, but Polygon policy deployment is not established by this release.

Area	Version 1 scope and boundary
Wallet	MetaMask is the current enabled wallet connection and authentication path. Other connectors remain outside the current launch path.
Assets	USDC and USDT are the current automation-eligible assets. WETH may be represented in portfolio diagnostics/reference views but is not currently eligible for automated policy execution. Arbitrary ERC-20 automation is not implied.
Diagnostics	HS-V1 engine, authenticated API, provider enrichment, and user interface are implemented in the live application and internally tested. Independent methodology validation, large-sample calibration, and external scientific review remain pending.
Scheduled Transfer	Configuration/review is implemented in the current application and the project reports a Base Mainnet contract deployment. Real-funds operational execution and independent audit remain unverified.
Basic Inheritance	Application flow is implemented and the project reports a Base Mainnet V2 contract deployment. Real-funds operational execution, legal succession treatment, and independent audit remain separate gates.
Refundable flow	Application, contract, API, poller/reconciliation, and internal E2E artifacts are reported, with a Base Mainnet escrow deployment. Independent audit, launch-market legal review, and sustained real-funds operation remain pending.
Authentication	MetaMask wallet connection, nonce issuance, wallet signature verification, session creation, protected dashboard access, and logout are implemented in the live application. Authentication does not authorize asset movement.
Check-in	Owner activity-confirmation UI/server logic is implemented and internally tested. A fully verified gasless Mainnet relayer/account-abstraction path remains Production verification pending.
Execution	Restricted executor, due-execution logic, dry-run controls, chain simulation, reconciliation, production-health, and monitoring components are present in current project evidence. Worker/Poller remain PARTIAL pending sustained Production runtime evidence.
Notifications	Application notification scheduling/templates and lifecycle hooks are implemented. External delivery reliability, retry/fallback operation, and regional availability remain not independently verified.
Custody	Scheduled Transfer and Basic Inheritance remain wallet-based until execution. Refundable Inheritance can introduce contract-controlled or escrow-like states, which are qualified separately.
Native token	Removed from Version 1; no current protocol need has been established.

Area	Version 1 scope and boundary
Subscription	Removed. The launch commercial model is execution-based.
Payment gateway	Not required for the approved fee model; fee logic is intended to settle from the transferred principal after successful execution.
Mainnet status	Base Mainnet contract addresses are publicly documented by Praxifi. Mainnet operational/real-funds launch is not claimed; Polygon policy deployment and independent smart-contract audit are not established.

16. Wallet Authentication

The current authentication flow is:

1. the user connects MetaMask;
2. the backend issues a unique nonce;
3. the user signs a scoped authentication message;
4. the backend verifies the signature, domain, nonce, and expiry as applicable;
5. the application creates a protected session.

The authentication signature proves control of the signing account for the session. It does not prove legal identity, legal ownership, capacity, or authority to act for another person. It does not transfer assets and does not grant token approval. Policy authorization and ERC-20 approval are separate actions [24].

The deployed authentication flow is internally tested, but independent penetration testing has not been completed. Ongoing Production review must address nonce reuse, origin and domain binding, session fixation, secure cookies, logout and revocation, replay, phishing, account switching, chain switching, CSRF, rate limiting, and monitoring.

17. Dashboard and User Mental Model

The current dashboard is organized around six user questions before exposing technical detail:

1. What assets does the connected wallet hold on supported networks?
2. What requires review in the current diagnostic?
3. What amount is protected or scheduled?
4. What is the next deadline or required action?
5. Can the owner still cancel or renew the policy?
6. What evidence confirms what happened?

The current user-facing navigation is:

- Analyze;
- Protect;
- Manage;
- Capital Allocation Research.

Developer terms such as Worker, Poller, RPC, ABI, nonce, allowance, and finality remain important in technical documentation, but should not be the primary language of the consumer workflow.

Historical Healthiness tracking is not a current Version 1 claim. It requires persisted snapshots, retention rules, methodology-version compatibility, recalculation logic, and privacy review.

18. Portfolio Healthiness Score

18.1 Purpose and classification

Portfolio Healthiness is a diagnostic and experimental scoring model intended to summarize selected observable characteristics of a supported portfolio. It is heuristic, versioned, non-predictive, non-advisory, and not a suitability assessment or universal measure of financial health.

Status field	Current statement
Model version	HS-V1
Model type	Experimental diagnostic model
Validation status	Independently unvalidated
Use limitation	Non-advisory and non-predictive; not a suitability assessment or credit rating
Calibration status	Weights, thresholds, and output rules include Version 1 design assumptions and heuristic calibration parameters

A higher result means only that the assessed evidence aligns more closely with the current methodology. It does not imply that the user should buy, sell, rebalance, or hold an asset. Improving a score does not necessarily improve future returns or welfare.

18.2 Current validation boundary

HS-V1 is implemented in the live application through a deterministic engine, authenticated API, provider enrichment, and user interface, with internal test and documentation artifacts reported by the project. Functioning implementation is distinct from scientific validation. The model has not completed:

- independent methodological review;
- peer review;
- large-sample calibration;
- prospective outcome testing;
- cross-market validation;
- regulatory suitability validation;
- or Production provider reliability testing.

The methodology therefore remains an experimental Product hypothesis even though the diagnostic is operational in the live application.

18.3 Diagnostic categories

The Version 1 model uses ten categories with weights totaling 10. The weights are **design assumptions**, informed by portfolio theory, digital-asset risk literature, and product objectives; they are not empirically established universal weights [3,6-10].

Category	Weight	Parameter classification	Version 1 purpose
Diversification	1.50	Adapted from established portfolio theory; implementation heuristic	Measures breadth and selected concentration/correlation signals.
Asset evidence quality	1.50	Product calibration parameter	Measures whether material assets are identifiable and supported by market, contract, issuer, and security evidence.
Concentration exposure	1.00	Adapted concept; heuristic thresholds	Identifies dominance by one or a small number of positions.

Category	Weight	Parameter classification	Version 1 purpose
Liquidity evidence	1.00	Adapted concept; product implementation	Evaluates selected market-depth and exit-capacity indicators where available.
Wallet and token security signals	1.50	Product calibration parameter	Reviews approvals, token warnings, and selected contract or issuer controls.
Stablecoin and reserve exposure	1.00	Product calibration parameter	Identifies dependence on stablecoins, issuers, counterparties, and cash-like reserves.
Protocol and smart-contract exposure	1.00	Product calibration parameter	Reviews identified protocol dependencies and available risk evidence.
Market volatility and drawdown evidence	0.75	Adapted financial risk concept	Summarizes selected historical market-risk indicators.
Leverage and liquidation exposure	0.50	Product calibration parameter	Applies only when supported leverage or collateral data is available.
Operational policy readiness	0.25	Product calibration parameter	Reviews whether supported balances, allowances, and policy evidence are operationally sufficient. It does not judge the user's character or discipline.

18.4 Signal ownership and double-counting

Each underlying signal should have one primary category. A secondary category may display the same fact for explanation, but should not automatically deduct the same risk twice.

Examples:

- asset weight belongs primarily to Concentration exposure;
- number and effective breadth of positions belong to Diversification;
- stablecoin issuer dependence belongs primarily to Stablecoin exposure;
- a token blacklist or pause signal belongs primarily to Wallet and token security;
- protocol TVL or identified protocol dependency belongs primarily to Protocol exposure;
- unsupported or missing data belongs to coverage and confidence, not automatically to a risk deduction.

The signal-to-category registry must be versioned and published with the methodology. A public methodology artifact remains pending.

18.5 Data sources

Potential Version 1 sources include:

- on-chain balances, token contracts, approvals, and transaction evidence;
- official chain and token information;
- market and historical data providers;
- DeFi protocol data providers;
- token-security providers;
- and internal policy-readiness data.

The current implementation uses CoinGecko, DefiLlama, GoPlus, and Etherscan V2 according to project implementation records. Provider presence does not prove accuracy or completeness. Production use requires versioned field mapping,

fallback behavior, rate-limit controls, freshness requirements, monitoring, and a process for provider disagreement. Independent provider validation remains pending.

18.6 Missing data and coverage

Missing evidence must not be converted into a zero risk score or treated as certainty. The output should show:

- full model weight;
- assessable weight;
- unavailable weight;
- supported portfolio value;
- unsupported exposure;
- coverage ratio;
- provider errors;
- and confidence.

The current minimum assessable weight of 6.0 is a **heuristic threshold**. It has not been empirically validated. A result may be produced only when critical categories and sufficient material exposure are covered. If security, concentration, or asset identification evidence is materially absent, the output should default to **Limited assessment** or **Data incomplete** even when the numerical coverage threshold is met.

The 6.0 threshold is a provisional Version 1 Product calibration parameter. It is not an independently validated boundary between a healthy and unhealthy portfolio.

18.7 Normalization

For categories with sufficient evidence, the provisional normalized score is:

Normalized score = 10 x weighted assessed contribution / assessable weight

This is a planned methodology rule. It must be interpreted with coverage and confidence. Normalization can otherwise overstate a portfolio when unassessed categories contain material risk.

18.8 Confidence

Confidence should reflect data coverage, freshness, provider consistency, asset identification, historical availability, and security-data availability. Confidence is not the probability that the score is correct.

No score should be presented without:

- methodology version;
- calculation timestamp;
- coverage ratio;
- confidence label;
- unavailable categories;
- and factor-level evidence.

18.9 Neutral diagnostic labels

V1.3 avoids labels such as Excellent, Healthy, Optimal, or Critical. Suggested output language includes:

- Data incomplete;
- Limited assessment;
- Elevated concentration;
- Moderate diversification;
- Broad diversification;
- Material unsupported exposure;

- Review recommended;
- No material issue detected in assessed evidence.

18.10 Token classification and stablecoins

Token classification must be versioned and evidence-based. Stablecoins should not automatically receive the same treatment as cash or deposits. Their risks may include issuer credit, reserve quality, redemption restrictions, blacklist or freeze authority, bridge structure, depeg risk, legal claims, and jurisdictional restrictions [9,10,13,31,32].

18.11 Model versioning

Every result should bind to a methodology version. A change to weight, threshold, provider mapping, signal ownership, token classification, or normalization should create a new version. Historical results from different versions must not be presented as directly comparable without recalculation or an explicit compatibility rule.

Methodology changes should require:

- a written rationale;
- approval by a disclosed product owner;
- research review by PraxiHub or another reviewer;
- security and privacy review where data use changes;
- versioned test cases;
- and user-facing release notes.

18.12 Validation plan

The proposed validation plan includes:

1. deterministic unit and property tests;
2. synthetic portfolio tests for edge cases;
3. provider-failure and stale-data tests;
4. backtesting against historical portfolio states without claiming causal outcomes;
5. sensitivity analysis for weights and thresholds;
6. comparison with independent risk indicators;
7. user-comprehension testing;
8. review for bias and unsupported assumptions;
9. external methodological review;
10. prospective testing after appropriate consent and legal review.

Until these steps are completed, the model remains experimental.

19. Scheduled Transfer

The current application supports Scheduled Transfer configuration and review, allowing the owner to define an automation-eligible token, gross amount, designated recipient, network, and future execution time. Project deployment records report a Base Mainnet Scheduled Transfer contract. The policy remains cancellable before execution under the applicable contract state; verified real-funds operation and independent audit are not claimed.

The current application flow is:

1. select a supported network and token;
2. enter the gross amount and recipient;
3. select date, time, and displayed timezone;
4. review the net amount, fee rate, fee cap, gas boundary, cancellation right, and risk;
5. grant the required approval;
6. authorize and activate the policy;

7. cancel before execution when needed;
8. after eligibility, allow the technical transaction executor to submit the call;
9. verify contract state, transaction receipt, and finality.

A scheduled time does not guarantee inclusion at an exact second. Network congestion, executor availability, gas policy, token controls, RPC conditions, or contract state may delay or prevent execution.

20. Basic Inheritance: Technical Continuity Policy

Basic Inheritance is the Product name for an inactivity-triggered technical transfer policy. The current application supports policy configuration and review, and Praxifi public documentation lists the Base Mainnet V2 policy-contract address. It does not create a will, identify a lawful heir, prove death, bypass mandatory succession rules, or guarantee that the designated recipient has legal entitlement.

The current application flow is:

1. select a supported network and token;
2. enter the gross amount;
3. enter and verify a designated recipient address;
4. choose an inactivity period;
5. configure reminder and activity-verification settings;
6. review the condition, cancellation right, fee, and legal limitation;
7. grant the required token approval;
8. authorize and activate the policy;
9. renew activity before the deadline or cancel before execution;
10. if no valid activity signal is received, permit the policy to become technically eligible.

The current application includes owner activity-confirmation logic. A valid check-in should renew the policy only after the system verifies the required owner authorization and replay protections. An application password may provide an additional factor, but it must not replace wallet authority. Password recovery is not currently supported and remains disabled.

21. Activity Confirmation

A genuinely gasless owner check-in cannot be created merely by calling a contract function that requires `msg.sender` to be the owner. A gasless design requires a signed instruction and a party that submits and pays for the on-chain transaction, or an account-abstraction mechanism [25,29].

The signed payload should bind at least:

- owner address;
- policy identifier or hash;
- chain ID;
- verifying contract;
- unique nonce;
- deadline;
- action type;
- and methodology or contract version where relevant.

EIP-712 improves the structure and readability of signed data, but it does not provide replay protection by itself [25]. Contract-side nonce consumption, expiry, chain/domain binding, and single-use enforcement remain necessary.

Status: Implemented and internally tested at the application layer. Production verification of the gasless Mainnet submission path remains pending; contract/deployment evidence for the underlying inactivity policy is disclosed separately in Appendix C.

22. Refundable Inheritance Lifecycle

Refundable Inheritance is a separately qualified continuity flow with recipient acceptance, owner refund, recipient waiver, and release states. It must not be described through one blanket non-custodial label.

22.1 Control lifecycle

Before escrow or activation: assets remain in the owner's wallet unless the implemented design requires a deposit. The user retains unilateral wallet control, subject to any approvals granted.

Contract-controlled or escrow-like state: if assets are transferred into a smart contract, the contract code rather than the platform or user wallet governs permitted movements. This can remain non-custodial in a narrow technical sense if no operator has discretionary withdrawal power, but the owner may have restricted unilateral access. Legal custody classification requires separate review.

Recipient acceptance: acceptance may establish a technical state or consent record. It does not by itself create inheritance rights or legal entitlement.

Refund: the owner may recover assets only under the conditions encoded in the relevant contract and lifecycle. The permitted caller, deadline, fee treatment, and evidence must be explicit.

Waiver: a designated recipient may submit a technical waiver if the contract supports it. A technical waiver may not waive legal rights under applicable law.

Release: assets may be released only when the required technical conditions are met. Release must not depend on discretionary operator selection of the recipient or amount.

Expiry and shutdown: the design must state whether the policy expires, how assets are recovered if the service shuts down, whether the contract remains callable without the Praxifi interface, and how a paused or upgraded system affects existing positions.

22.2 Current status

Current project records include the Refundable Inheritance application flow, contract implementation, acceptance/refund/waiver/release APIs, event polling, reconciliation, and internal E2E/runtime evidence; Praxifi public documentation also lists the Base Mainnet Refund Escrow address. Independent smart-contract audit, final public admin disclosure, launch-jurisdiction legal review, and sustained real-funds operational verification remain pending.

Figure 1 shows the control-state distinction that remains relevant in Product interfaces and legal analysis. Technical acceptance does not establish legal beneficiary rights. Mainnet deployment does not remove the need to verify escrow administration, shutdown, upgrade, recovery, and real-funds operational controls before treating the lifecycle as Production-verified.

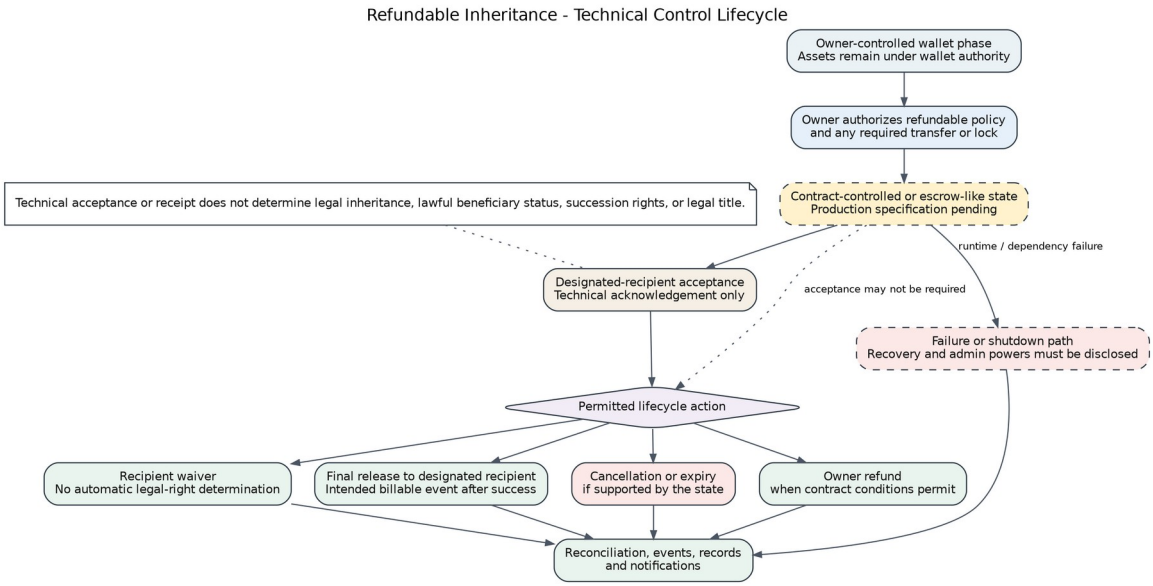

Figure 1. Refundable Inheritance technical control lifecycle and legal-right boundary.

Status: Mainnet deployed on Base / implemented and internally tested / Production operational verification and independent audit pending.

22.3 Fee-triggering event

The 1% fee capped at USD 49 applies only to the defined successful billable event. Acceptance, waiver, refund, expiry, or an incomplete lifecycle must not be assumed billable without an explicit rule. Current project code and tests report successful release to the designated recipient as the billable event; real-funds Production settlement remains not independently verified.

23. Policy Lifecycle

State	Meaning
Draft	No execution authority exists.
Awaiting authorization	Wallet signature, token approval, or required contract action is incomplete.
Active	The policy is authorized and its condition is monitored.
Approaching eligibility	The deadline is near; reminders and readiness checks are active.
Eligible	The technical condition permits an execution attempt. Eligibility is not a guarantee of success.
Submitted	A transaction has been submitted but is not yet final.
Confirming	The system is waiting for the required finality and reconciliation.
Executed	The intended one-time action is final according to the configured finality rule.
Cancelled	The owner cancelled before execution. Token approval may remain and may require separate revocation.
Failed	An attempt failed; the retry, expiry, and notification rules apply.
Expired	The policy is no longer valid.
Paused	Execution is temporarily blocked by an authorized safety control.

State	Meaning
Disputed or inconsistent	Application records and chain evidence do not agree and require investigation.

Figure 2 separates authentication, policy authorization, token approval, eligibility, transaction submission, contract enforcement, and post-execution evidence. Wallet login alone does not authorize asset movement.

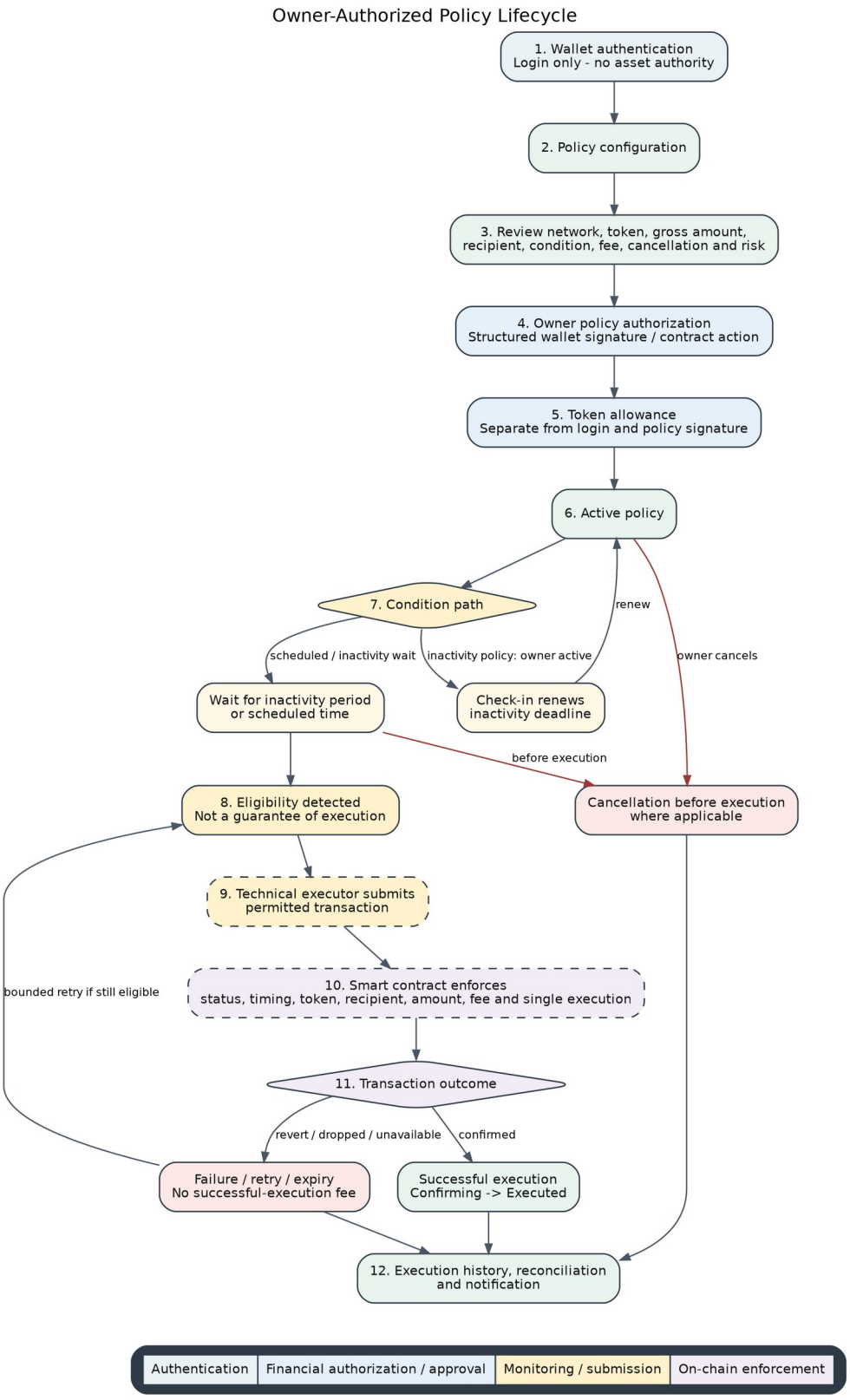

Figure 2. Owner-authorized policy lifecycle and separation of authentication, approval, eligibility, and execution.

24. Notifications

Notifications are operational aids, not the source of truth. The current application includes notification scheduling/templates and lifecycle hooks for upcoming check-ins, scheduled deadlines, low balance, insufficient allowance, token warnings, failed transactions, successful execution, refundable lifecycle changes, and security events.

Critical status must be verified against signed records, contract state, and blockchain evidence. Delivery failure must be monitored and visible. Regional availability of notification channels may vary, and no single channel should be assumed reliable.

PART IV - Technical Architecture

25. Architecture Overview

The current Version 1 architecture flow is:

User Wallet -> Praxifi Interface -> Authentication and Policy Review -> Signed Policy and Token Approval -> Monitoring and Readiness -> Technical Transaction Executor -> Smart Contract Validation -> Token Contract -> Blockchain Evidence -> Reconciliation and Notification

Figure 3 summarizes the Version 1 system and its main trust points. The diagram distinguishes user-controlled components, Praxifi-operated services, external providers, and blockchain components. Dashed borders identify components whose full Production operational evidence remains gated. Base Mainnet deployment of a contract does not by itself remove those operational or audit gates. The diagram does not imply decentralization.

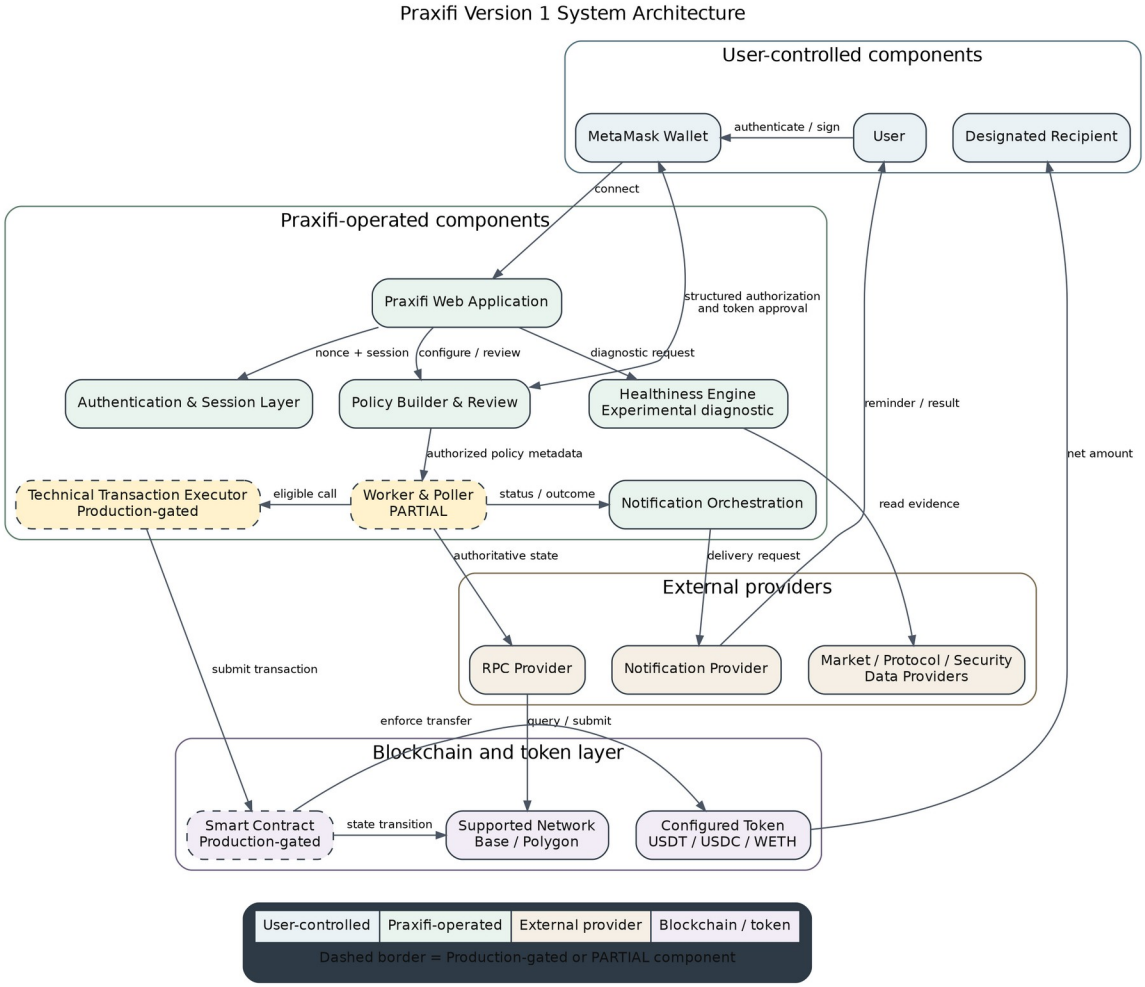

Figure 3. Praxifi Version 1 system architecture and trust boundaries.

26. System Components

User wallet

Signs authentication and policy actions and grants token approval. Praxifi must not receive or reconstruct the private key.

Frontend

Collects intent, displays the gross and net amount, explains risk, and presents the signed policy summary. A compromised frontend can mislead the user; signed payload visibility and independent verification remain important.

Backend

Issues nonces, manages sessions, stores non-authoritative metadata, coordinates data providers, monitors policies, and supports notifications and reconciliation. It should fail closed when required evidence is unavailable.

Portfolio diagnostic services

Collect balances and selected market, token, issuer, protocol, and security evidence. Provider results are inputs, not authority over assets.

Smart contracts

The current policy contracts are designed to enforce policy authorization, chain and token scope, recipient, amount, eligibility, cancellation, execution status, fee logic, caller restrictions, and single execution. Project deployment records report Base Mainnet deployments for PraxifiInactivityExecutorV2, PraxifiScheduledTransfer, and PraxifiRefundEscrow; addresses are listed in Appendix C. Independent smart-contract audit is not completed, and sustained real-funds operational behavior is not independently verified.

Technical transaction executor

Submits eligible transactions. It should not select the token, recipient, amount, or economic objective. Its availability, key security, gas funding, and replacement process remain operational dependencies.

Worker and Poller

The Worker identifies candidate policies and prepares or submits permitted operations. The Poller verifies chain state, receipts, finality, and lifecycle events. Current project evidence includes fail-closed dry-run controls, worker-chain simulation, refundable event polling and reconciliation, runtime E2E artifacts, production-health checks, and monitoring/alerting modules. Because sustained Production operation, failover performance, service-level objectives, and real-funds runtime have not been independently verified, the combined Worker/Poller status remains PARTIAL.

Status: PARTIAL.

Reconciliation

Compares application metadata, signed policy evidence, contract state, token events, escrow events, and transaction receipts. A mismatch must be exposed, not silently overwritten.

27. Initial Policy Model

The fields below describe the intended policy schema. They are an architecture model, not a public ABI or deployment record.

Identity and authorization

Field	Purpose
policyId	Unique identifier.
policyType	Scheduled Transfer, Basic Inheritance, or separately deployed Refundable flow.
owner	Authorizing account. This is a technical role, not a legal-title determination.
chainId	Prevents cross-chain ambiguity and supports signature domain binding.

Field	Purpose
token	Configured token contract.
designatedRecipient	Owner-selected destination. Not necessarily a lawful beneficiary.
nonce or policyHash	Binds owner authorization and prevents silent modification.
createdAt	Creation time.
version	Contract and policy schema version.

Economic and timing parameters

Field	Purpose
grossAmount	Maximum amount the policy may debit or release.
feeRateBps	Fee rate accepted when the policy is authorized.
feeCapUsd	USD-denominated cap accepted under the fee version.
feeVersion	Identifies the pricing and calculation rules.
executionTime	Scheduled eligibility time, when applicable.
inactivitySeconds	Owner-selected inactivity period, when applicable.
lastCheckInAt	Most recent valid renewal time.
expiry	Optional policy expiry.

Lifecycle controls

Field	Purpose
cancelled	Blocks later execution.
executed	Enforces one-time completion.
paused	Blocks execution under the defined safety authority.

Principal operations

- create or authorize policy;
- check in through a direct or signed flow;
- cancel policy;
- execute eligible policy;
- record failure or reconciliation status where appropriate;
- refundable-flow acceptance, refund, waiver, and release when separately supported.

Expected rejections

- zero or invalid address;
- unsupported network or token;
- zero or dust amount;
- invalid fee version;
- unauthorized caller;
- expired or replayed signature;
- chain/domain mismatch;
- execution before eligibility;
- execution after cancellation, expiry, or completion;
- duplicate submission that violates idempotency;
- insufficient balance or allowance;

- token pause, blacklist, or transfer failure;
- stale or unavailable required oracle evidence;
- and a fee or net amount below permitted bounds.

28. Worker, Poller, and Execution Boundaries

Candidate selection

The Worker must read authoritative contract state before submission. Application status alone is insufficient.

Idempotency

A policy intended for one-time execution requires a unique idempotency key or contract state that prevents duplicate economic execution even if the Worker retries.

Finality

A submitted transaction should remain in a confirming state until the configured chain-specific finality threshold is met. The application must handle replacement transactions, dropped transactions, and chain reorganization.

Retry

Retries should be bounded by policy status, gas policy, token state, expiry, and duplicate protection. A failed attempt must not silently change the fee or recipient amount.

Monitoring requirements

Production release requires monitoring for:

- Worker and Poller health;
- RPC health and inconsistency;
- executor balance;
- gas treasury thresholds;
- pending, failed, replaced, and duplicate transactions;
- delayed finality;
- reconciliation mismatch;
- token pause or blacklist events where observable;
- provider failure;
- key rotation;
- and emergency pause status.

Internal operational and runtime test artifacts are reported; sustained Production operational verification remains pending.

29. Trust Boundaries

The system is not trustless. It separates authority among components, but each component introduces a distinct failure or centralization risk.

User and application layer

Component	Permitted role	Trust or centralization risk
User wallet	Sign and approve	Compromise or loss can defeat user-side security.
Frontend	Display and collect intent	Can misrepresent parameters or direct users to malicious approvals.
Backend	Sessions, metadata, monitoring	Can censor, delay, or display stale information.

Component	Permitted role	Trust or centralization risk
Notification providers	Deliver reminders	Delivery, privacy, regional-access, and censorship risk.

Execution and data layer

Component	Permitted role	Trust or centralization risk
Data providers	Market, protocol, security evidence	Can be stale, incomplete, inconsistent, or unavailable.
RPC providers	Chain access	Can fail, lag, censor, or provide inconsistent views.
Technical transaction executor	Submit eligible transactions	Key compromise, censorship, downtime, or gas shortage.
Worker and Poller	Detect, submit, reconcile	Logic error or outage may delay or misclassify status.
Smart contracts	Enforce programmed state	Bugs, upgrade authority, and token interaction risk.

External and administrative layer

Component	Permitted role	Trust or centralization risk
Stablecoin issuer	Token administration	Pause, freeze, blacklist, redemption, or policy intervention.
Admin or deployer	Configure or change system	Centralized authority, compromise, or governance failure.
Fee recipient	Receive service fee	Accounting, tax, sanctions, and key-management risk.

30. Data and Evidence Boundaries

Critical financial authority should not exist only in a mutable database. The database may index and explain policy state, but the owner authorization, material parameters, cancellation, execution, and transaction evidence should be independently verifiable through signatures, contract state, and blockchain events.

Each material implementation statement should link to a public or reviewable artifact when available:

- source-code location;
- commit;
- test report;
- deployment record;
- contract address;
- audit report;
- governance address;
- or architecture decision.

Project source, test, deployment, and operations artifacts are internally recorded and some deployment facts are disclosed in Appendix C. Where an artifact has not been independently reproduced or publicly verified, the correct evidence label remains internal report rather than independent verification.

31. Fee Calculation and Execution Boundaries

Launch fee schedule

Policy	Fee rate	Fee cap
Scheduled Transfer	0.5%	USD 3
Basic Inheritance	0.75%	USD 25
Refundable Inheritance	1.0%	USD 49

The approved model has no subscription and no separate payment gateway. The intended service fee is execution-based and deducted from the transferred principal after successful execution.

Planned accounting invariant

Gross amount = net recipient amount + service fee

Current project code and internal tests implement fee-calculation and settlement logic for the supported policy paths. The accounting invariant remains subject to real-funds Production verification, token-specific behavior, and final governance disclosure.

Calculation

For a policy with gross token amount G, token price in USD P, fee rate r, and USD cap C:

1. gross USD value = $G \times P$;
2. uncapped fee USD = $\text{gross USD value} \times r$;
3. charged fee USD = $\min(\text{uncapped fee USD}, C)$;
4. fee token amount = $\text{charged fee USD} / P$;
5. net recipient token amount = $G - \text{fee token amount}$.

Current automation eligibility is limited to USDC and USDT; WETH and other non-stable assets are not currently eligible for automated policy execution. For USD-denominated stablecoins, a one-token-equals-one-dollar assumption must not be used blindly during a depeg. The Production pricing policy must define the source, allowed deviation, freshness, fallback, and suspension behavior.

Required Production specification

- exact pricing source and contract/feed address;
- decimal normalization;
- rounding direction;
- minimum gross amount;
- dust handling;
- stale-price threshold;
- provider or oracle failure behavior;
- depeg behavior;
- token-specific fee transfer behavior;
- fee recipient address and governance;
- failed execution behavior;
- recordkeeping and receipt format;
- and tax uncertainty.

Internal fee tests are reported. The final public Production pricing/oracle specification, real-funds settlement verification, and fee-recipient governance disclosure remain incomplete.

Rounding and user protection

Rounding must not cause the contract to debit more than the authorized gross amount. The user should see the maximum fee and estimated net amount before authorization. The executed fee should be reproducible from the accepted fee version and final pricing evidence.

Failed execution

No service fee should be charged when the intended policy action fails and no recipient transfer occurs. Gas or third-party costs may still arise and must be disclosed separately.

Refundable lifecycle

The current implementation defines successful final release to the designated recipient as the intended billable event for Refundable Inheritance. Acceptance, waiver, refund, cancellation, expiry, and failed release are not assumed billable. Real-funds Production settlement remains not independently verified.

32. Upgrade and Dependency Gates

An upgrade, module, or external dependency should not be presented as available merely because a general ecosystem page lists a network. Production use requires the exact module version, repository release, deployment address, audit status, compatibility test, and governance path.

Safe modules can execute powerful actions and may create account-takeover risk if malicious or incorrectly configured [29]. Recovery research therefore remains blocked until a specific deployment and control model is verified.

Active policies require an explicit migration policy. An upgrade must not silently rewrite the recipient, amount, fee version, or condition. Users should be informed of material changes and retain a safe cancellation or exit path where technically possible.

PART V - Security, Threat Model, and Operational Resilience

33. Security Principles

Praxifi's security objective is not to eliminate risk. It is to restrict authority, make state changes observable, and provide safe failure and intervention paths.

Least privilege

Each policy should authorize only the configured chain, token, gross amount, designated recipient, condition, and execution path.

Owner-authorized material parameters

A technical transaction executor or routine operator should not be able to change the owner, recipient, token, amount, fee version, or condition. A material change should require a new owner authorization and policy hash.

Cancellation and renewal

A pending policy should remain cancellable or renewable according to its lifecycle. Cancellation of the policy does not necessarily revoke the underlying token allowance; the interface must explain this distinction.

Single execution and idempotency

A one-time policy must become permanently non-executable after successful completion. Off-chain retry logic must not create duplicate economic execution.

Transparent evidence

Creation, authorization, check-in, cancellation, eligibility, submission, finality, failure, and execution should produce verifiable evidence appropriate to the state.

Non-custodial is not risk-free

A wallet-based design can reduce platform custody risk while leaving wallet compromise, approval risk, contract risk, stablecoin issuer risk, infrastructure dependence, and legal uncertainty unresolved.

34. Formal Threat Model

The tables below form an initial threat model, not an independent audit. Evidence status is limited to internal reports and design documentation unless otherwise stated. A risk is not considered eliminated unless the relevant control has been implemented, tested, monitored, and independently reviewed where appropriate.

User, authentication, and interface risks

Threat or failure	Actor or component	Existing or intended mitigation	Residual risk	Responsibility	Status or evidence
Owner-wallet compromise	External attacker / wallet	Wallet signature, visible policy summary, exact approval guidance	Attacker with wallet authority may create, cancel, or redirect user-approved actions	User and wallet provider	Unresolved; outside platform control
Wallet loss	User / wallet	Continuity policies established before loss; recovery research	Praxifi does not recover the private key; owner may lose cancellation ability	User	Accepted limitation
Recipient-address error	User / frontend	Address validation, checksum, network context, review step, optional warning	Valid but wrong address may be irreversible	User and frontend	Reduced, not eliminated

Threat or failure	Actor or component	Existing or intended mitigation	Residual risk	Responsibility	Status or evidence
Malicious frontend	Attacker / frontend	Signed structured data, exact parameter review, policy hash, CSP and supply-chain controls	Wallet may display incomplete information; user may approve malicious spender	Operator and user	Independent security review pending.
Session hijacking	Attacker / backend	Secure cookies, session rotation, nonce use, origin binding, logout	Off-chain metadata or actions may be exposed; wallet signature still required for financial authority	Operator	Independent penetration testing pending.
Password compromise	Attacker / application	Hashing, rate limits, lockout, monitoring; password not sole authority	Brute force, reuse, social engineering	Operator and user	Partially designed
Signature replay	Attacker / relayer	Nonce, deadline, chain ID, verifying contract, action type, single-use consumption	Incorrect domain or nonce logic can permit replay	Contract and backend	Internal test evidence reported; independent reproduction pending.
EIP-712 domain error	Developer / contract	Chain-bound domain, contract address, versioned schema	Wrong domain may enable cross-contract or cross-chain replay	Contract developer	Independent security review pending.
Chain-ID mismatch	User / frontend / relayer	Network validation and signed chain ID	Wrong-network approval or execution attempt	Frontend, backend, contract	Architecture validated

Execution, contract, and infrastructure risks

Threat or failure	Actor or component	Existing or intended mitigation	Residual risk	Responsibility	Status or evidence
Executor-key compromise	External attacker / executor	Restricted caller, scoped contract authority, managed key, rate limit, revoke and replacement	Attacker may submit every eligible call and consume gas; upgrade/admin weaknesses may enlarge impact	Operator	Production controls pending
Executor censorship	Executor/operator	Monitoring, retry, replacement path, transparent eligibility	Delayed or absent execution	Operator	Unresolved liveness risk
Executor downtime	Infrastructure	Health checks, balance alerts, failover plan	Eligible policies may not execute on time	Operator	PARTIAL
Poller failure	Poller	Independent receipt/state checks, retries, monitoring	Stale application state or premature finality	Operator	PARTIAL
RPC failure	RPC provider	Multiple providers, timeouts, circuit breakers, retries	Delay or inability to verify state	Operator/provider	Production setup pending
RPC inconsistency	RPC provider	Cross-provider comparison for critical state	Incorrect eligibility or finality view	Operator/provider	Sustained Production operational verification pending.

Threat or failure	Actor or component	Existing or intended mitigation	Residual risk	Responsibility	Status or evidence
Chain reorganization	Network	Confirmation threshold and reconciliation	Previously observed transaction may disappear or change position	Operator/network	Finality spec pending
Insufficient finality	Operator	Chain-specific finality rule	Premature executed status	Operator	Sustained Production finality verification pending.
Smart-contract vulnerability	Contract	Least privilege, tests, review, audit, pause/exit design	Loss, lock, or unauthorized movement	Developer/operator	No independent audit
Approval misuse	Contract/attacker	Exact or bounded allowance, spender verification, revocation guidance	Persistent approval may remain after cancellation	User/contract/ operator	Reduced, not eliminated
Unlimited allowance	User/frontend	Do not request unlimited approval for one-time policy	User may grant broader authority through another interface	User/frontend	Policy requirement; verification pending
Insufficient balance	User wallet	Readiness monitoring and warning	Execution failure	User	Accepted / visible
Gas shortage	Executor treasury	Minimum-balance alerts and funding process	Delayed execution	Operator	Monitoring pending
MEV or front-running	Network/validator	No user-selectable price action in V1; idempotency and bounds	Timing and gas effects; public eligibility may be observed	Network/operator	Context-dependent
Timestamp dependence	Block producer/network	Conservative windows and no exact-second guarantee	Small timestamp variation around eligibility	Contract/network	Requires test
Notification failure	Provider	Redundant channels and visible delivery state	Missed check-in or deadline	Operator/provider/ user	Reduced, not eliminated

Token, pricing, and diagnostic-data risks

Threat or failure	Actor or component	Existing or intended mitigation	Residual risk	Responsibility	Status or evidence
Issuer intervention	Stablecoin issuer	Diversification and disclosure	Freeze, redemption, or policy changes	Issuer/user	Accepted external risk
Decimal or rounding error	Contract/token	Safe fixed-point arithmetic, token-decimal registry, invariant tests	Dust loss or excess debit	Contract developer	Independent property/invariant verification pending.
Oracle manipulation	Oracle/provider	Trusted feed selection, freshness, deviation checks, pause	Incorrect USD fee	Operator/oracle	Final Production pricing specification pending.
Stale pricing	Oracle/provider	Timestamp and freshness threshold	Incorrect fee during depeg or volatility	Operator/oracle	Pending

Threat or failure	Actor or component	Existing or intended mitigation	Residual risk	Responsibility	Status or evidence
Data-provider poisoning	Provider/attacker	Multiple sources, confidence, provenance, no execution authority	Misleading diagnostic	Operator/provider	Reduced through disclosure
Incomplete portfolio data	Chain/provider	Coverage and unsupported exposure output	Overconfident score	Operator/user	Methodology control

Administrative, operational, and privacy risks

Threat or failure	Actor or component	Existing or intended mitigation	Residual risk	Responsibility	Status or evidence
Admin-key compromise	Admin	Multisig, timelock, least privilege, monitoring	Pause, upgrade, executor, or configuration abuse	Operator	Actual controls undisclosed
Deployer-key compromise	Deployer	Retire or restrict deployer authority	Unauthorized deployment or upgrade if authority remains	Operator	Final public admin-control disclosure pending.
Service shutdown	Operator	Contract-level exit/cancel paths and public runbook	Users may lose interface, monitoring, or executor	Operator/user	Public shutdown/exit-plan verification pending.
Dependency compromise	Package/provider	Pin versions, SBOM, review, monitoring	Frontend, backend, or contract compromise	Operator/provider	Independent supply-chain review pending.
Privacy leakage	Operator/provider	Data minimization, access control, retention and provider mapping	Wallet may be linked to identity, IP, or behavior	Operator/provider	Formal launch privacy review pending.

Other identified risks

Threat or failure	Actor or component	Existing or intended mitigation	Residual risk	Responsibility	Status or evidence
Password recovery abuse	Attacker / application	Recovery disabled in current version	User cannot recover the password through Praxifi	Operator	Not supported
Worker logic failure	Worker	Dry-run, fail-closed design, deterministic checks, tests	Missed or incorrect candidate selection	Operator	PARTIAL; internal report only
Token-specific behavior	Token contract	Allowlist configured contracts; token adapters and tests	Fee-on-transfer, rebasing, non-standard return, upgrade, pause	Token issuer/operator	Public token-behaviour matrix verification pending.
Stablecoin pause or blacklist	Issuer/token	Pre-execution state checks and disclosure	Issuer can block transfer despite policy eligibility [31,32]	Issuer/user	Transferred external risk
Insufficient allowance	User wallet	Allowance monitoring and warning	Execution failure	User	Accepted / visible

Threat or failure	Actor or component	Existing or intended mitigation	Residual risk	Responsibility	Status or evidence
Gas-price spike	Network	Gas ceiling, delay and retry policy	Execution delay or economic failure	Operator/user	Policy pending
Fee calculation error	Contract/oracle	Versioned formula, tests, max gross debit	Wrong net amount or fee	Contract/operator	Internal fee tests reported; real-funds verification pending.
Gas-treasury compromise	Treasury	Separate key, balance limit, monitoring	Loss of gas funds and liveness	Operator	Live treasury-control verification pending.
Upgrade authority abuse	Admin/proxy	Immutable design or disclosed multisig/timelock	Rules or authority may change after activation	Operator	Architecture unresolved
Pause authority abuse	Admin	Narrow pause scope and public events	Censorship or indefinite lock	Operator	Final public pause-control disclosure pending.
Cross-chain assumption	Developer/user	No cross-chain execution in Version 1; explicit chain scope	Wrong asset/address assumptions	Operator/user	Reduced by scope

The model should be updated whenever the architecture, supported token set, executor design, fee mechanism, provider set, or administrative authority changes. Independent security review pending. Sustained Production operational verification pending.

35. Executor Security and Liveness

The executor should be a replaceable transaction-delivery component. Production controls should include:

- protected key storage;
- on-chain caller restrictions;
- balance and gas alerts;
- rate limits and anomaly detection;
- transaction simulation;
- queue and nonce management;
- replacement-transaction policy;
- failover authorization;
- transparent executor-change events;
- and a tested emergency revoke procedure.

Changing the executor must not change an active policy’s recipient, token, amount, condition, or fee version. The party authorized to replace the executor is a material governance actor and must be disclosed.

36. Emergency and Incident Response

Incident response should align with a documented Govern, Identify, Protect, Detect, Respond, and Recover process [20,21]. A minimum runbook should cover:

1. detect and classify the incident;
2. stop new execution attempts through the narrowest safe control;
3. revoke or disable a compromised executor;

4. preserve evidence and logs;
5. identify policies approaching eligibility;
6. notify affected users through available channels;
7. publish cancellation, approval-revocation, or exit guidance;
8. reconcile application, contract, token, and chain state;
9. restore services through an approved process;
10. publish a post-incident report and remediation status.

Production release requires incident ownership, communication channels, escalation thresholds, service objectives, and a responsible-disclosure process. Production incident exercise evidence pending.

37. Audit, Bug Bounty, and Security Evidence

This public release does not claim an independent smart-contract audit or Production penetration test. Before representing the system as verified for sustained real-funds operation, Praxifi should publish or maintain reviewable evidence for:

- the exact source-code commit;
- build and compiler settings;
- contract addresses and deployment transactions;
- unit, integration, invariant, and adversarial test reports;
- audit scope and findings;
- remediation status;
- admin and upgrade configuration;
- monitoring and incident runbooks;
- and bug-bounty or responsible-disclosure details.

PART VI - Legal and Regulatory Uncertainty

38. Legal Status of This Section

No governing jurisdiction or final regulatory classification has been established for the purposes of this whitepaper.

This section is a jurisdiction-neutral, function-based risk framework. It identifies facts that may affect classification. International standards, principles, and guidance are not automatically binding national law. Regional and national materials are cited only as comparative examples.

Praxifi's regulatory classification cannot be determined solely from its non-custodial design. Classification may depend on the functions actually performed, the degree of operational control, the execution model, the fee structure, the jurisdictions involved, and the manner in which the service is offered.

39. Nature and Scope of the Technical Service

Praxifi is designed to provide software interfaces, portfolio diagnostics, policy creation, monitoring, transaction submission, reconciliation, and notifications. Some functions are informational; others may participate in initiating or submitting on-chain transactions.

The legal analysis must therefore examine actual behavior rather than product labels such as protocol, automation, infrastructure, self-custody, or non-custodial.

40. Functional Regulatory Perimeter

Potentially relevant functions include:

- software provision;
- wallet authentication;
- transaction initiation and submission;
- automated execution;
- smart-contract design and administration;
- custody or safeguarding during any contract-controlled state;
- crypto-asset transfer services;
- payment-related services;
- portfolio diagnostics;
- investment recommendation risk;
- stablecoin-related activity;
- data processing;
- sanctions and AML controls;
- fee collection;
- and cross-border availability.

Insufficient evidence exists to determine Praxifi's regulatory classification. This issue requires review under the laws applicable to the operator, users, service activities, and relevant transactions.

FATF materials provide international AML/CFT standards and recommendations that generally require domestic implementation; they are not themselves a global operating licence [12,13]. FSB, IOSCO, OECD, and BIS materials identify policy and risk questions but do not determine Praxifi's legal status [14-19].

41. Technical Control and Legal Ownership

Blockchain control, practical access, and legal entitlement are distinct. Private-key control can enable a transaction but does not necessarily establish lawful ownership. Loss of a key can impair disposition without automatically extinguishing rights. Receipt of a token can change technical control without resolving competing claims [11].

The terms owner and ownership are used in technical sections to identify the account that authorizes a policy. They do not state a final legal conclusion.

42. Technical Execution and Legal Entitlement

A smart contract can enforce programmed conditions. It cannot independently determine whether:

- the person authorizing the rule had legal capacity;
- the asset was subject to another person's rights;
- a recipient is a lawful heir;
- a transfer violates a court order, sanctions rule, insolvency process, tax duty, matrimonial property rule, fiduciary duty, or mandatory succession law;
- or the recipient may lawfully retain the asset.

Technical execution does not necessarily determine legal entitlement.

43. Digital-Asset Transfer Limitations

An on-chain transfer may be irreversible at protocol level while remaining legally contestable. Token contracts may pause, blacklist, upgrade, rebase, charge fees, or otherwise behave differently from a standard ERC-20. Networks can experience reorganization, congestion, or outages. The platform does not guarantee final legal or economic settlement.

44. Inheritance and Succession Limitations

"Inheritance" is used as a product-category description for an inactivity-triggered technical transfer policy. It does not determine legal inheritance, identify lawful heirs, establish succession rights, or replace a will, trust, court process, estate administrator, fiduciary, or other legally recognized arrangement.

Inactivity does not prove death, incapacity, abandonment, identity, or legal intent. A designated recipient is not necessarily a lawful beneficiary. A technical transaction executor is not necessarily an executor of an estate.

Succession rules differ materially across jurisdictions and may include mandatory shares, formal wills, probate or administration, conflict-of-law rules, taxes, creditor rights, family claims, and restrictions on contractual transfers. Comparative instruments demonstrate this variation but do not govern Praxifi by default [33,34].

Before offering continuity products as a fully reviewed launch service in a specific market, Praxifi requires a separate jurisdiction memorandum addressing the operator, target users, technical flow, fee recipient, contractual terms, and succession treatment. Launch-jurisdiction legal review is not completed by this whitepaper.

45. Custody and Safeguarding Considerations

Basic Inheritance and Scheduled Transfer are designed so that assets remain in the user's wallet until execution. This may reduce traditional platform custody risk, but it does not automatically establish a legal non-custodial classification.

Refundable Inheritance may include a contract-controlled or escrow-like state. Relevant questions include:

- whether the user retains unilateral withdrawal rights;
- whether an admin can redirect, pause, upgrade, or release assets;
- whether the contract can be changed;
- whether the platform controls required keys;
- and whether users depend on the operator for exit.

These facts may affect custody, safeguarding, fiduciary, insolvency, and consumer-protection analysis.

46. Payment and Transfer-Service Exposure

Scheduled token transfers, technical transaction submission, fee deduction, and execution on behalf of users may be relevant to payment or transfer-service classifications in some jurisdictions. The result may depend on whether Praxifi merely provides software, controls execution, receives client instructions, collects fees, holds assets, or can intervene in settlement.

Non-custodial design alone is not an exemption. Insufficient evidence exists to determine classification.

47. Crypto-Asset Service Classification Uncertainty

International and regional frameworks increasingly regulate activities such as custody, transfer, execution of orders, operating trading platforms, exchange, advice, and portfolio management. Whether Praxifi performs a regulated crypto-asset service depends on the legal definition and actual operation in each market.

The main whitepaper does not assume that Praxifi is a VASP, CASP, money transmitter, payment service, or exempt software provider. Each possibility requires analysis after the launch structure is known.

48. Investment-Service and Advice Boundaries

The Healthiness diagnostic is intended to be informational, non-predictive, and non-advisory. However, labels, rankings, recommendations, prompts, policy drafts, and personalization can affect whether a service is treated as advice or another regulated investment activity.

Praxifi should avoid:

- personalized buy, sell, or rebalance instructions in Version 1;
- suitability claims;
- return forecasts;
- guaranteed improvement language;
- or automatic portfolio execution based only on the diagnostic.

A future recommendation or allocation feature requires a new legal and product review.

49. AML, CFT, and Sanctions Considerations

The applicable obligations may depend on whether Praxifi is classified as a regulated service provider, the parties it serves, its control over transactions, the jurisdictions involved, and the stablecoins or infrastructure used.

Possible requirements may include customer due diligence, sanctions screening, transaction monitoring, recordkeeping, reporting, geographic restrictions, and Travel Rule obligations. FATF guidance and updates identify relevant risks but require implementation through domestic or regional law [12,13].

A sanctions or AML control can itself create user, privacy, false-positive, and censorship risks. The launch design must document what screening occurs, who provides it, what happens after a match, and how users can seek review. Launch-jurisdiction legal and regulatory review remains pending.

50. Stablecoin and Issuer-Control Risks

USDT and USDC are not equivalent to sovereign cash or insured deposits. Their issuers or contracts may have pause, freeze, blacklist, redemption, or compliance controls. The user’s ability to transfer may be affected even when the Praxifi policy is otherwise eligible [31,32].

Stablecoin rules, reserve requirements, marketing restrictions, and service-provider obligations differ by jurisdiction. No global classification is assumed.

51. Consumer-Protection Considerations

Material disclosures should be presented before authorization, not hidden in appendices. Users should understand:

- what Praxifi can and cannot move;
- the exact gross amount, maximum fee, and estimated net amount;
- the cancellation deadline;
- that approval may remain after policy cancellation;
- that execution is not guaranteed;
- that inactivity has no legal meaning by itself;
- that stablecoin and contract risks remain;
- and that the user is responsible for the recipient address and wallet security.

Error, complaint, refund, shutdown, and incident processes may be required even when the software is non-custodial.

52. Data Protection and Privacy

A public wallet address is pseudonymous, not necessarily anonymous. Portfolio data may reveal wealth, behavior, protocol use, counterparties, and personal relationships. Linking a wallet to a session, IP address, contact information, designated recipient, or notification channel increases privacy risk.

The Production privacy program should disclose:

- data collected;
- purpose and lawful basis where required;
- retention periods;
- access controls;
- processors and providers;
- cross-border processing;
- deletion or restriction processes;
- analytics;
- wallet-session linkage;
- incident response;
- and whether data is used for AI training.

Portfolio and policy data should not be used for advertising or model training without a clear legal basis and explicit disclosure. NIST privacy materials offer a voluntary risk-management framework; they do not establish legal compliance [22]. A formal launch privacy review remains pending.

53. Tax and Reporting Uncertainty

Policy creation, fee collection, transfer, receipt, refund, waiver, and release may have tax or reporting consequences. OECD CARF is an international tax-transparency framework that depends on implementation by participating jurisdictions; it is not automatically binding everywhere [16,17].

Users remain responsible for obtaining advice and records appropriate to their circumstances. Praxifi must determine whether the final operator or service is subject to reporting, invoicing, withholding, recordkeeping, or information-exchange duties.

54. Cross-Border Availability

A blockchain interface may be technically accessible from many locations without being lawfully offerable in all of them. Availability may depend on the operator’s entity, target market, user location, transaction parties, stablecoin terms, sanctions, licensing, consumer law, data processing, and tax rules.

No global availability or legality is claimed. Restricted jurisdictions and launch markets have not been established in this whitepaper.

55. Independent Professional Advice

Users considering continuity or inheritance-related policies should obtain independent legal, tax, estate-planning, and financial advice appropriate to their jurisdiction and circumstances. Technical execution should be coordinated with legally recognized documents and processes where required.

56. No Global Compliance Representation

This whitepaper describes the technical design of Praxifi, known legal uncertainties, and areas of potential regulatory exposure. It does not determine Praxifi’s legal classification, establish compliance in any jurisdiction, or replace jurisdiction-specific legal, regulatory, or tax advice.

A future Launch Jurisdiction Memorandum should be prepared only after confirming:

- the legal entity and place of incorporation;
- operational locations;
- initial target and restricted markets;
- launch functions;
- contract deployment and upgrade model;
- fee recipient;
- execution-service structure;
- data-processing structure;
- custody boundaries;
- governing terms;
- and local mandatory law.

PART VII - Economics, Governance, and Roadmap

57. Business Model

The launch model is execution-based. Creating, reviewing, cancelling, or checking in does not itself carry a Praxifi service fee. A service fee is intended only after the defined successful execution event.

The approved launch schedule is:

- Scheduled Transfer: 0.5%, capped at USD 3;
- Basic Inheritance: 0.75%, capped at USD 25;
- Refundable Inheritance: 1%, capped at USD 49.

Blockchain gas, relayer funding, third-party data costs, taxes, and other external costs are distinct from the service fee. Whether Praxifi subsidizes, passes through, or recovers any of those costs must be disclosed before launch.

The model does not require a subscription, native token, or separate payment gateway. Each policy should preserve the fee version accepted by the owner so that later pricing changes do not silently affect an active instruction.

58. No Native Token

A Praxifi token is not part of Version 1. Rule creation, authorization, fee payment, and governance do not currently require a native token. Creating one without a necessary protocol function would add speculative, legal, liquidity, governance, and security complexity.

59. Governance and Administrative Controls

This public release does not claim that the final Production multisig, timelock, pause, upgrade, or executor-control configuration has been independently verified.

Before a fully verified Production launch, Praxifi should publish a governance matrix covering:

Authority	Required disclosure
Contract deployer	Address, retained powers, and retirement plan.
Upgrade authority	Whether contracts are immutable or upgradeable; implementation-change process.
Pause authority	Operations affected, conditions, duration, and public events.
Executor administrator	Who can add, remove, or replace the executor.
Fee administrator	Whether rates, caps, oracle, or recipient can change.
Methodology owner	Who approves Healthiness changes and version releases.
Data-provider administrator	Who can add or remove evidence providers.
Incident authority	Who can activate emergency controls and communicate incidents.

Sensitive authority should use appropriately designed multisignature and delay controls where practical, but the final configuration must be disclosed rather than assumed.

60. Roadmap

Phase 1 - Public Release and Evidence Maintenance

- maintain the Version 1 scope, Public Release terminology, and status register;
- maintain versioned source, test, deployment, and Mainnet evidence appropriate to each public claim;
- continue authenticated browser QA and regression checks against the live application;

- maintain Base Mainnet contract evidence and complete any future Polygon deployment-specific verification before claiming Polygon policy availability;
- publish the final Production fee/pricing specification and fee-recipient governance;
- complete independent smart-contract/security review plus launch privacy and legal reviews;
- publish the final Production governance, incident, and administrative-control configuration.

Phase 2 - Mainnet Operational Verification

- verify sustained real-funds Scheduled Transfer operation on the supported Mainnet path;
- verify sustained real-funds Basic Inheritance operation on the supported Mainnet path;
- complete token-specific approval and issuer-control verification for the supported automation assets;
- verify owner cancellation and allowance-state behavior against deployed contracts;
- verify Production execution history, finality, replacement, and reconciliation behavior;
- verify notification delivery, retry/fallback, and operational monitoring;
- exercise Worker/Poller monitoring, failover, incident response, and service-level objectives;
- and verify real-funds execution-fee settlement and records.

Phase 3 - Refundable Continuity Hardening

- finalize and independently review the deployed escrow/control specification;
- acceptance, refund, waiver, release, and expiry;
- shutdown and exit paths;
- complete sustained multi-process Production validation;
- audit and legal review.

Phase 4 - Financial Policy Engine

- recurring transfers;
- multi-policy coordination;
- asset reservations;
- priority and conflict rules;
- institutional research and templates;
- portfolio alerts without automatic investment execution.

Phase 5 - Advanced Portfolio Intelligence and Capital Allocation Research

- methodology calibration;
- external review;
- historical diagnostics with version compatibility;
- user-authorized policy drafting;
- rule-based allocation research;
- risk budgets and liquidity constraints;
- legal and security gates before any execution capability.

61. Development Gates

No phase should advance solely because the previous phase is described in a roadmap. Release requires:

- technical correctness;
- understandable consent;
- security testing;
- operational monitoring;
- incident readiness;
- privacy review;
- legal and regulatory review;

- and public evidence appropriate to the claim.

PART VIII - Risks, Positioning, and Conclusion

62. Principal Risks

Technical

Smart-contract vulnerabilities, incorrect authorization, token incompatibility, chain reorganization, RPC failure, frontend compromise, dependency compromise, upgrade risk, signature replay, fee calculation, oracle failure, and reconciliation errors.

Operational

Executor or Worker outage, insufficient gas treasury, delayed finality, notification failure, provider failure, incomplete monitoring, incorrect configuration, and delayed incident response.

User-side

Wallet compromise, lost access, wrong recipient, misunderstood approval, failure to check in, and reliance on a diagnostic beyond its intended purpose.

Financial

Asset volatility, stablecoin depeg, issuer freeze, liquidity limitations, transaction costs, protocol failure, and a gross amount that becomes insufficient after fees or market movement.

Legal and regulatory

Custody or transfer classification, succession disputes, consumer protection, data protection, AML/CFT, sanctions, tax, reporting, licensing, cross-border availability, and disagreement between technical execution and legal entitlement.

Research and model

Healthiness bias, unstable provider data, unvalidated thresholds, incomplete chain coverage, unsupported assets, false confidence, and misuse as advice.

63. Product Limitations

Praxifi does not:

- recover or recreate private keys;
- verify death, incapacity, abandonment, identity, or legal competence;
- identify lawful heirs;
- guarantee legal inheritance;
- guarantee execution;
- eliminate wallet, token, contract, issuer, infrastructure, or regulatory risk;
- provide investment management in Version 1;
- guarantee portfolio improvement or returns;
- support every network, wallet, token, protocol, or smart-account standard;
- claim independent validation of internal implementation;
- claim that Mainnet contract deployment is equivalent to an audited, sustained real-funds Production launch;
- or establish legal compliance in any jurisdiction.

64. Category and Positioning

Praxifi’s intended category is **non-custodial financial-policy infrastructure for digital wealth**, with the qualification that regulatory classification depends on actual functions and jurisdiction.

Its potential differentiation is the combination of:

- experimental portfolio diagnostics;
- technical continuity policies;
- Scheduled Transfer;
- narrow authorization;
- owner cancellation;
- executor separation;
- evidence and reconciliation;
- and a research pathway toward broader policy coordination.

This is a positioning thesis, not a claim of global uniqueness. A maintained competitor and prior-art register is required for stronger claims.

65. Long-Term Vision

The long-term vision is a financial operating layer in which users coordinate selected digital-asset decisions through understandable policies. Wallets or smart accounts remain the authorization layer. Smart contracts enforce defined constraints. Executors deliver eligible transactions. Praxifi coordinates diagnostics, policy creation, monitoring, evidence, and user experience. PraxiHub provides a research process that should be capable of challenging the assumptions behind that system.

The vision does not require unrestricted automation. A mature system should make it easier to see what may happen, why it may happen, who authorized it, what can stop it, and what evidence confirms the result.

66. Conclusion

Praxifi began with research into wealth management and developed around a practical question: how can a selected financial decision remain understandable and operational after the moment in which it was made?

Version 1 is intentionally limited. The web application is live, and Portfolio Healthiness is operational as an experimental diagnostic rather than investment advice. Praxifi public documentation lists Base Mainnet contract addresses for Scheduled Transfer, Basic Inheritance, and Refundable Escrow, but publication of those addresses is not a guarantee of execution, legal entitlement, independent audit, or sustained real-funds operation. Refundable Inheritance remains a separately qualified lifecycle with additional operational and legal gates.

The product’s credibility depends less on ambitious language than on visible boundaries. The user should know who can act, what can move, how much can move, what condition permits it, what it costs, how it can be cancelled, and what remains uncertain.

Analysis should inform. The owner should authorize. Contracts should constrain. Execution should be evidenced. Legal entitlement should remain a separate question for applicable law.

Final Publication Disclaimer

This public whitepaper describes Praxifi’s research foundations, live application status, intended technical architecture, Product boundaries, Mainnet deployment status, and current development evidence. It does not constitute legal, tax, or investment advice; establish regulatory compliance; determine legal inheritance; establish an independent smart-contract audit; prove sustained real-funds Production operation; or establish independent scientific validation of the Healthiness Score. Application availability, on-chain deployment, Product operation, and legal treatment are separate states and may differ according to development status and applicable requirements.

APPENDICES

Appendix A - Terminology

Term	Definition
Owner	The account that authorizes a policy. This is a technical role and not necessarily a final determination of legal ownership.
Designated recipient	The address selected by the owner. It is not necessarily a lawful heir or beneficiary.
Technical transaction executor	The restricted service or account that submits an eligible transaction. It is not an executor of an estate.
Financial policy	A versioned instruction defining an action, condition, constraints, cancellation or expiry rules, and evidence.
Continuous Ownership	A Praxifi-defined design principle concerning continuity of owner-authorized control instructions. It does not establish or transfer legal title.
Digital Asset Continuity	Research and infrastructure addressing continued practical access, transaction capability, and owner-authorized instructions across disruption.
Basic Inheritance	Product label for an inactivity-triggered technical continuity policy. It does not determine legal inheritance.
Refundable Inheritance	Product label for a separately qualified contract lifecycle involving acceptance, refund, waiver, and release.
Check-in	A valid owner-authorized activity confirmation that renews an inactivity deadline. It does not prove life, capacity, or identity.
Exact allowance	An ERC-20 allowance limited to the intended amount. Token behavior and remaining approval must still be monitored.
Portfolio Healthiness	Experimental, heuristic, versioned diagnostic of selected portfolio evidence. It is not advice, suitability, rating, or return prediction.
Coverage	The portion of the methodology and portfolio value supported by sufficient evidence.
Confidence	A data-quality indicator, not a probability of correctness.
Non-custodial design	A design intended to avoid unrestricted operator possession or discretion. Legal classification requires functional analysis.
Production verification	Evidence that deployment, monitoring, security, governance, operations, and release controls are ready for real use.

Appendix B - Product Truth and Implementation Status

The status register reflects the August 2026 public-release state. It separates live application behavior, internal implementation/test evidence, Mainnet deployment evidence, Production operational verification, and independent review. Internal reports are not independent verification.

Access, networks, assets, and diagnostics

Capability	Status	Evidence boundary
Praxifi Web Application	Live	Founder-confirmed public deployment; implemented web components are functioning. Public website availability is not Mainnet/Production evidence.

Capability	Status	Evidence boundary
MetaMask authentication	Operational	Nonce/signature/session/protected-dashboard routes and current deployed application flow are internally evidenced; independent penetration testing pending.
Base Mainnet	Mainnet deployed	Praxifi public documentation lists the current Base Mainnet policy-contract addresses. Sustained real-funds operation and independent audit remain separately unverified.
Polygon PoS	Architecture validated / Production verification pending	Approved Version 1 network scope; no current policy-contract deployment is established in this release.
USDC and USDT	Current automation scope	Automation eligibility is limited to these stablecoins under the current Product policy. Token/issuer controls remain external risks.
WETH	Not currently supported for automation	May be represented in diagnostics/reference views; automated policy execution is blocked for the current launch path.
Healthiness engine and UI	Operational / implemented and internally tested	Live HS-V1 diagnostic implementation is reported. Independent methodology validation, calibration, and scientific review remain pending.

Policy and continuity flows

Capability	Status	Evidence boundary
Scheduled Transfer	Mainnet deployed on Base / application flow operational	Public contract-address documentation and internal contract/app test evidence are available. Real-funds operational execution and independent audit are not independently verified.
Basic Inheritance	Mainnet deployed on Base / application flow operational	Internal V2 deployment/test evidence reported. Legal inheritance not determined; real-funds operation and independent audit remain separate.
Check-in	Implemented and internally tested at application layer	Owner activity-confirmation logic exists. A fully verified gasless Mainnet submission path remains Production verification pending.
Password-assisted check-in	Implemented application layer	May provide an additional application factor; it cannot replace wallet authority. Password recovery remains not supported.

Capability	Status	Evidence boundary
Refundable Inheritance	Mainnet deployed on Base / internally E2E tested	Contract/app/API/poller/reconciliation evidence reported. Independent audit, launch legal review, and sustained real-funds operation remain pending.
Exact allowance	Implemented and internally tested	Approval/readiness logic is reported; current automation scope is USDC/USDT and issuer/token behavior remains an external dependency.
Cancellation	Implemented and internally tested	Policy cancellation and ERC-20 allowance revocation remain distinct actions; deployed-path Production behavior should continue to be monitored.

Execution and operations

Capability	Status	Evidence boundary
Executor	Implemented and internally tested	Executor registry/wallet/execution verification artifacts are reported. Sustained Production liveness, key management, replacement, and real-funds operation are not independently verified.
Worker/Poller	PARTIAL	Dry-run, simulation, event polling/reconciliation, runtime E2E, and monitoring artifacts are reported; sustained Production runtime/failover/SLO evidence remains incomplete.
Execution history	Operational at application level	Current application presents execution/audit history; full Production reconciliation evidence remains not independently verified.
Notifications	Implemented at application level	Scheduling/templates/hooks are reported; external delivery reliability, fallback, and regional monitoring remain pending.
Monitoring and health checks	Implemented and internally tested	Production-health, ops-monitoring and alerting modules/tests are reported. Live operational alert performance is not independently verified.
Fee settlement	Implemented and internally tested / Production verification pending	Fee logic/tests are reported. Real-funds settlement, final pricing governance, and fee-recipient disclosure remain not independently verified.

Research, future scope, and exclusions

Capability	Status	Evidence boundary
Social recovery	Research-stage / blocked dependency	Not part of the current launch path; exact module/version/deployment/audit/compatibility required before use.
Private-key recovery	Not currently supported	Rejected from Version 1.
Multi-wallet	Future version	MetaMask is the current enabled wallet path.
Portfolio automation	Future version	No automatic investment execution based on Healthiness in Version 1.
Capital Allocation Research	Research-stage	No Product or return claim.
Native token	Removed	No current protocol need.
Subscription	Removed	Execution-based fee model selected.
Payment gateway	Not currently required	Fee model is designed around execution-based deduction rather than a separate gateway.
Polygon Mainnet policy deployment	Unable to independently verify	No Polygon policy-contract deployment is established in the current release evidence.
Mainnet real-funds operation	Unable to independently verify	Base contract deployment is reported, but sustained real-funds Product operation is not claimed by this release.
Independent smart-contract audit	Independent validation pending	No independent audit of the exact deployed scope is claimed.

Appendix C - Evidence and Release Artifacts

This appendix records implementation and release evidence that is relevant to public claims. “Internally reported” means project-controlled source, tests, deployment reports, or founder confirmation exists, but independent verification may remain unavailable. Mainnet deployment is not treated as equivalent to an independent audit or sustained Production operation.

- Source/version evidence: the project reports current code and deployment/QA artifacts in its repository; public independent reproduction of the latest commit state was not completed during this release review.
- Contract ABI evidence: generated/tested interfaces are reported internally; a stable public ABI package is not independently verified in this release.
- Base Mainnet contract addresses: PraxifiInactivityExecutorV2 - 0x29c6efF6D4f2687568e77E6CAA34102C92b07552; PraxifiRefundEscrow - 0x4A0f710cEE41E1987E0f5B6601Bf1Fd229d78b82; PraxifiScheduledTransfer - 0x35874Fd2bF16d0C7D2430BcBAae5D10547F83A56. These addresses are listed in Praxifi public documentation. This first-party publication is not treated as independent audit evidence or proof of sustained real-funds operation.
- Deployment evidence: project deployment reports and evidence files record Base Mainnet deployment and revalidation. Independent review of every deployment artifact was not completed for this publication.
- Testing evidence: contract, server, UI, Worker, fee, monitoring, and runtime E2E tests/reports are present in current project records. Independent reproduction is not complete.

- Property/invariant evidence: internal tests are reported; independent property/invariant verification of the exact deployed scope is not complete.
- Token-behaviour evidence: current automated policy scope is restricted to USDC and USDT. A public comprehensive token-behaviour matrix is not independently verified.
- Independent smart-contract audit: not completed or claimed for the exact deployed scope as of this release.
- Security review: internal security-review and threat-model artifacts are reported; independent security review remains pending.
- Independent penetration test: not completed for this release.
- Finality specification: chain/reconciliation logic is internally documented and tested; sustained Production finality behavior is not independently verified.
- Operational test evidence: runtime E2E, production-health, monitoring, dry-run and simulation artifacts are reported; sustained real-funds operation remains unverified.
- Incident response: internal incident-response and emergency-response documentation is reported; Production exercise evidence is not independently verified.
- Administrative control disclosure: final public Production multisig/timelock/upgrade/executor authority configuration is not independently verified.
- Pause/emergency controls: implementation and internal tests are reported; final public Production authority configuration remains incomplete.
- Treasury/gas controls: internal treasury and operations policy artifacts are reported; live operational control verification is not independently available.
- Pricing/oracle specification: current automation is limited to USDC/USDT; final public depeg/pricing/freshness policy is not independently verified.
- Fee tests: internal fee-calculation and settlement tests are reported; real-funds Production settlement is not independently verified.
- Fee recipient/governance: final public Production fee-recipient and change-control disclosure remains incomplete.
- Healthiness methodology artifact: HS-V1 methodology is described in this whitepaper; the versioned implementation registry and independent validation package remain pending.
- Provider validation: current data providers are integrated, but independent Production reliability and disagreement/fallback validation remain pending.
- Privacy review: a formal launch privacy review has not been completed by this whitepaper.
- Legal review: a launch-jurisdiction memorandum has not been completed by this whitepaper.
- Supply-chain review: independent review of application/contract dependencies and build provenance remains pending.

Appendix D - Research and Evidence Method

This edition uses a structured narrative evidence review rather than claiming a complete systematic review. Sources are selected from peer-reviewed research, official standards, protocol documentation, institutional reports, legal instruments, and project evidence.

Each future PraxiHub source register should record:

- search date and scope;
- database or source;
- title, author, institution, and publication date;
- version and access date;
- source type;
- claim supported;
- limitations;
- conflict of interest;
- review status;
- and correction history.

A source that supports a general problem does not certify Praxifi's implementation. Internal reports must be labeled as internal until code, tests, deployment records, or external review are available.

Appendix E - Comparative Jurisdictional Examples

The examples below are not the governing law of Praxifi and do not determine its classification.

Comparative jurisdictional example - United States model law: RUFADAA addresses fiduciary access to digital assets through a model-law approach adopted or adapted by jurisdictions. It illustrates that access, disclosure, account terms, and fiduciary authority may be governed by formal legal processes [33].

Comparative jurisdictional example - European Union succession: Regulation (EU) No 650/2012 provides a regional framework for jurisdiction, applicable law, recognition, and certificates in cross-border succession. It illustrates the complexity of succession and conflict-of-law analysis [34].

Comparative jurisdictional example - European Union crypto services: MiCA and the Transfer of Funds Regulation define regional categories and obligations for certain crypto-asset services and transfers. They do not apply globally and do not determine Praxifi's classification without factual and territorial analysis [35,36].

Detailed analysis belongs in a future Launch Jurisdiction Memorandum after the operating entity, target markets, restricted jurisdictions, fee recipient, custody boundaries, and execution model are confirmed.

Appendix F - Disclosures

- This document is technical and informational.
- It does not provide investment, legal, tax, estate-planning, custody, regulatory, or cybersecurity advice.
- Praxifi does not guarantee security, execution, inheritance, portfolio improvement, or returns.
- Blockchain transactions may be irreversible and may fail because of user, wallet, token, issuer, contract, network, executor, provider, or infrastructure conditions.
- Users are responsible for wallet security, recipient accuracy, policy parameters, approvals, and obtaining advice appropriate to their circumstances.
- Availability may vary by jurisdiction.
- Future capabilities may change or be removed after research, testing, audit, legal review, or operational assessment.

Appendix G - References and Evidence Base

The source set was structurally rechecked for this Public Release. Hyperlinks use descriptive labels and clean canonical URLs. Access dates shown in individual entries record the prior verification date. International standards, principles, recommendations, guidance, comparative laws, technical standards, academic research, issuer materials, and commercial provider documentation are identified by source type and are not treated as automatically binding law or independent proof of Praxifi effectiveness.

1. Nakamoto, S. (2008). Bitcoin: A Peer-to-Peer Electronic Cash System. Technical whitepaper. Direct source: Bitcoin whitepaper - official PDF. Accessed 4 August 2026.
2. Buterin, V. (2014). A Next-Generation Smart Contract and Decentralized Application Platform. Technical whitepaper. Direct source: Ethereum whitepaper. Accessed 4 August 2026.
3. Markowitz, H. (1952). "Portfolio Selection." The Journal of Finance, 7(1), 77-91. DOI: 10.1111/j.1540-6261.1952.tb01525.x. Peer-reviewed research. Direct source: Portfolio Selection - DOI record. Accessed 4 August 2026.
4. D'Acunto, F., Prabhala, N., & Rossi, A. G. (2019). "The Promises and Pitfalls of Robo-Advising." The Review of Financial Studies, 32(5), 1983-2020. DOI: 10.1093/rfs/hhz014. Peer-reviewed research. Direct source: The Promises and Pitfalls of Robo-Advising - DOI record. Accessed 4 August 2026.
5. Rossi, A. G., & Utkus, S. (2024). "The Diversification and Welfare Effects of Robo-Advising." Journal of Financial Economics, 157, 103869. Peer-reviewed research. Direct source: The Diversification and Welfare Effects of Robo-Advising - publisher page. Accessed 4 August 2026.

6. Liu, Y., & Tsyvinski, A. (2021). “Risks and Returns of Cryptocurrency.” The Review of Financial Studies, 34(6), 2689-2727. DOI: 10.1093/rfs/hhaa113. Peer-reviewed research. Direct source: Risks and Returns of Cryptocurrency - DOI record. Accessed 4 August 2026.
7. Amihud, Y. (2002). “Illiquidity and Stock Returns.” Journal of Financial Markets, 5(1), 31-56. Peer-reviewed research. Direct source: Illiquidity and Stock Returns - DOI record. Accessed 4 August 2026.
8. Schär, F. (2021). “Decentralized Finance: On Blockchain- and Smart Contract-Based Financial Markets.” Federal Reserve Bank of St. Louis Review, 103(2), 153-174. Institutional research. Direct source: Federal Reserve Bank of St. Louis Review article. Accessed 4 August 2026.
9. Bank for International Settlements. (2023). The Crypto Ecosystem: Key Elements and Risks. Institutional report. Direct source: BIS report: The Crypto Ecosystem. Accessed 4 August 2026.
10. Aramonte, S., Huang, W., & Schrimpf, A. (2021). “DeFi Risks and the Decentralisation Illusion.” BIS Quarterly Review. Institutional research. Direct source: BIS Quarterly Review: DeFi Risks. Accessed 4 August 2026.
11. UNIDROIT. (2023). Principles on Digital Assets and Private Law. International principles; not automatically binding national law. Direct source: UNIDROIT Principles on Digital Assets and Private Law. Accessed 4 August 2026.
12. Financial Action Task Force. (2021). Updated Guidance for a Risk-Based Approach to Virtual Assets and VASPs. International standards and guidance requiring domestic implementation. Direct source: FATF Updated Virtual Assets and VASPs Guidance. Accessed 4 August 2026.
13. Financial Action Task Force. (2026). Seventh Targeted Update on Implementation of the FATF Standards on Virtual Assets and VASPs. International implementation review. Direct source: FATF 2026 Targeted Update on Virtual Assets and VASPs. Accessed 4 August 2026.
14. Financial Stability Board. (2023). Global Regulatory Framework for Crypto-Asset Activities. International recommendations; not directly binding law. Direct source: FSB Global Regulatory Framework for Crypto-Asset Activities. Accessed 4 August 2026.
15. International Organization of Securities Commissions. (2023). Policy Recommendations for Crypto and Digital Asset Markets. International recommendations. Direct source: IOSCO Policy Recommendations for Crypto and Digital Asset Markets. Accessed 4 August 2026.
16. OECD. (2023). International Standards for Automatic Exchange of Information in Tax Matters: Crypto-Asset Reporting Framework. International tax framework requiring implementation. Direct source: OECD Crypto-Asset Reporting Framework. Accessed 4 August 2026.
17. OECD. (2025). CARF Monitoring and Implementation Update. Institutional implementation material. Direct source: OECD tax-transparency implementation materials. Accessed 4 August 2026.
18. IMF & FSB. (2023). Synthesis Paper: Policies for Crypto-Assets. International policy report. Direct source: IMF-FSB Synthesis Paper on Crypto-Asset Policies. Accessed 4 August 2026.
19. BIS. (2025). Cryptocurrencies and Decentralised Finance: Functions and Financial Stability Implications. Institutional research. Direct source: BIS paper on cryptocurrencies and decentralised finance. Accessed 4 August 2026.
20. NIST. (2024). Cybersecurity Framework 2.0, NIST CSWP 29. Voluntary cybersecurity framework. Direct source: NIST Cybersecurity Framework 2.0. Accessed 4 August 2026.
21. NIST. (2025). Incident Response Recommendations and Considerations for Cyber Risk Management, SP 800-61 Rev. 3. Guidance. Direct source: NIST SP 800-61 Rev. 3. Accessed 4 August 2026.
22. NIST. Privacy Framework. Voluntary privacy-risk framework. Direct source: NIST Privacy Framework. Accessed 4 August 2026.
23. Ethereum Improvement Proposals. ERC-20 Token Standard. Technical standard. Direct source: ERC-20 Token Standard. Accessed 4 August 2026.
24. Ethereum Improvement Proposals. ERC-4361: Sign-In with Ethereum. Technical standard. Direct source: ERC-4361 Sign-In with Ethereum. Accessed 4 August 2026.
25. Ethereum Improvement Proposals. EIP-712: Typed Structured Data Hashing and Signing. Technical standard. Direct source: EIP-712 Typed Structured Data Specification. Accessed 4 August 2026.
26. Ethereum Improvement Proposals. ERC-2612: Permit Extension for EIP-20 Signed Approvals. Technical standard. Direct source: ERC-2612 Permit Extension. Accessed 4 August 2026.

27. Base Documentation. Connecting to Base; Base Mainnet chain ID 8453. Official technical documentation. Direct source: Base Mainnet connection documentation. Accessed 4 August 2026.
28. Polygon Documentation. Polygon PoS RPC Endpoints; mainnet chain ID 137. Official technical documentation. Direct source: Polygon PoS RPC documentation. Accessed 4 August 2026.
29. Safe Documentation. Safe Modules. Official technical documentation warning that modules can execute powerful transactions and may create security risk. Direct source: Safe Smart Account Modules documentation. Accessed 4 August 2026.
30. OpenZeppelin. Contracts Documentation: Access Control, Pausable, Reentrancy, and ERC-20 Utilities. Official technical documentation. Direct source: OpenZeppelin Contracts documentation. Accessed 4 August 2026.
31. Circle. Stablecoin EVM Contracts. Issuer source describing administrative token controls; product/issuer evidence, not independent assessment. Direct source: Circle stablecoin EVM contract repository. Accessed 4 August 2026.
32. Tether. Legal Terms. Issuer source describing freeze and service rights; product/issuer evidence. Direct source: Tether legal terms. Accessed 4 August 2026.
33. Uniform Law Commission. Revised Uniform Fiduciary Access to Digital Assets Act. Comparative jurisdictional model law; not governing law of Praxifi. Direct source: Uniform Law Commission RUFADAA materials. Accessed 4 August 2026.
34. European Union. Regulation (EU) No 650/2012 on succession. Comparative regional example; not governing law of Praxifi. Direct source: EU Succession Regulation - comparative example. Accessed 4 August 2026.
35. European Union. Regulation (EU) 2023/1114 on markets in crypto-assets. Comparative regional example; not a global standard. Direct source: EU Markets in Crypto-Assets Regulation - comparative example. Accessed 4 August 2026.
36. European Union. Regulation (EU) 2023/1113 on information accompanying transfers of funds and certain crypto-assets. Comparative regional example. Direct source: EU Transfer of Funds Regulation - comparative example. Accessed 4 August 2026.
37. Černevičienė, J., & Kabašinskas, A. (2024). “Explainable Artificial Intelligence in Finance: A Systematic Literature Review.” Artificial Intelligence Review, 57, 216. DOI: 10.1007/s10462-024-10854-8. Peer-reviewed research. Direct source: Explainable Artificial Intelligence in Finance - DOI record. Accessed 4 August 2026.
38. CFA Institute Research and Policy Center. (2025). Explainable AI in Finance: Addressing the Needs of Diverse Stakeholders. Institutional research. Direct source: CFA Institute report, Explainable AI in Finance: Addressing the Needs of Diverse Stakeholders. Accessed 4 August 2026.
39. Luu, L., Chu, D.-H., Olickel, H., Saxena, P., & Hobor, A. (2016). “Making Smart Contracts Smarter.” ACM CCS. Peer-reviewed security research. Direct source: Making Smart Contracts Smarter - DOI record. Accessed 4 August 2026.
40. Atzei, N., Bartoletti, M., & Cimoli, T. (2017). “A Survey of Attacks on Ethereum Smart Contracts.” POST. Peer-reviewed security research. Direct source: Survey of Attacks on Ethereum Smart Contracts - DOI record. Accessed 4 August 2026.
41. Hernando-Corrochano, J., Pastor-Vargas, R., & Hernández-Berlinches, R. (2025). “Trusted Wills for Digital Assets Using Blockchain: A Practical Case.” Blockchain: Research and Applications, 6(3), 100289. Peer-reviewed research; does not certify Praxifi. Direct source: Trusted Wills for Digital Assets Using Blockchain - DOI record. Accessed 4 August 2026.
42. Kraiwanit, T., Limna, P., & Suradinkura, S. (2025). “Digital Asset Adoption in Inheritance Planning: Evidence from Thailand.” Journal of Risk and Financial Management, 18(6), 330. Jurisdiction-specific empirical research; not governing law. Direct source: Digital Asset Adoption in Inheritance Planning - DOI record. Accessed 4 August 2026.
43. CoinGecko. API documentation. Commercial data-provider documentation; supports only provider capability descriptions. Direct source: CoinGecko API documentation. Accessed 4 August 2026.
44. DefiLlama. API documentation. Commercial/open data-provider documentation. Direct source: DefiLlama API documentation. Accessed 4 August 2026.
45. GoPlus Security. Token Security API documentation. Commercial provider documentation. Direct source: GoPlus Token Security API documentation. Accessed 4 August 2026.
46. Etherscan. API V2 documentation. Commercial provider documentation. Direct source: Etherscan API V2 documentation. Accessed 4 August 2026.

Appendix H - Version 1.3 Material Changes

Version 1.3:

- separates technical control from legal ownership;
- defines inheritance as a qualified product label;
- replaces the public Excess Return label with Capital Allocation Research;
- classifies Healthiness weights and thresholds as heuristics or product parameters;
- removes authoritative Healthiness bands;
- adds a validation plan and missing-data controls;
- marks Worker and Poller status PARTIAL;
- explains Refundable Inheritance control states;
- adds fee, oracle, rounding, and billable-event boundaries;
- adds a formal threat model and trust-point register;
- introduces a function-based, jurisdiction-neutral legal section;
- operationally qualifies PraxiHub independence;
- and adds visible evidence-status disclosures rather than inventing technical proof.

synchronizes Version 1.3 with the founder-confirmed live web application state;

records publicly documented Base Mainnet contract addresses while separating deployment status from audited Production operation;

updates authentication and Healthiness language from proposed/internal-only wording to current live-application wording where supported;

restricts current automation eligibility to USDC and USDT and keeps WETH outside automated policy execution;

retains Worker/Poller as PARTIAL because sustained Production runtime evidence remains incomplete;

replaces internal evidence placeholders with publication-quality evidence-status disclosures;

and changes document status from Publication Candidate to Version 1.3 Public Release without changing the core architecture, fee model, legal position, or scientific qualification.