

# Bedrock Capabilities Statement

Digital-asset tax and accounting engine for institutions. Built for CFOs, controllers, tax professionals, and their auditors: every number traceable, every action attributed, every report reproducible.

**One governed record for digital asset tax, accounting, reconciliation, and audit support.**

|                                                                  |                                                                          |
|------------------------------------------------------------------|--------------------------------------------------------------------------|
| <b>Pillar 01:</b> Tax Engine, Options, OTC, and Lot Intelligence | <b>Pillar 02:</b> Ingestion, Matching, Pricing, and Data Integrity       |
| <b>Pillar 03:</b> Activity Coverage and Classification           | <b>Pillar 04:</b> Reconciliation, Transparency, and Explainability       |
| <b>Pillar 05:</b> GAAP Books, Fair Value, Reports, and Exports   | <b>Pillar 06:</b> Controls, Access, Administration, and Platform Support |

---

## Tax Engine, Options, OTC, and Lot Intelligence

Bedrock keeps tax treatment tied to the lots, elections, counterparty movements, rules, and decisions that produced the result.

### Tax Engine & Cost Basis

- Full spot lot tracking: every acquisition becomes a lot with immutable identity, tracked from purchase to disposition
- Cost basis, proceeds, and fee allocation computed under current Treasury regulations (Treas. Reg. §1.1001-7 and TD 10000)
- Swap-fee decomposition: fees paid in the disposed token, the acquired token, or split across multiple sell legs are handled correctly
- §1091 wash sale detection with automatic basis adjustment and disallowed-loss carry
- §1092 straddle identification with holding-period reset and loss deferral
- §1223 holding-period tacking across transfers: lots keep their original acquisition dates when assets move between wallets
- Short-term vs. long-term character determined per lot, per disposition
- Charitable donation treatment with zero-gain election
- Retroactive lot designation: designate specific lots after the fact, with full holding-period isolation
- Configurable taxpayer tax policy per profile

### Options & Derivatives

- §1234(b) options income treatment for written options (covered calls, puts)
- Options position lifecycle: write, close, exercise, expire, buyback
- Straddle detection with automated suggestions and multi-leg straddle identification
- Physically-settled option delivery tied back to specific underlying lots
- Options import from exchange files plus manual entry for bespoke paper and OTC contracts
- Settlement reconciliation: confirms option settlements against on-chain movements
- Lot encumbrance tracking: know which lots are pledged as option backing at any moment
- Backing-lot assignment with smart-pick suggestions ranked by tax outcome

### OTC & External Transfer Management

- External Transfer Rules: designate an address as a counterparty you settle through, and sends to it are recorded as transfers, not sales
- Captured value moves to a dedicated tracking ledger, one per asset per counterparty, carrying original cost basis and acquisition dates intact: no gain or loss at send time
- Record the disposition when the trade actually settles, not when the value left your wallet, with gain or loss computed against that original basis and holding period
- Automatic return capture: value the counterparty sends back unsold comes home as a transfer with basis restored, so a round trip nets to zero tax events; anything beyond the tracked position books as an ordinary deposit at market value and is flagged for review
- Retroactive application with a full preview: see which transactions a rule captures and how gain/loss changes before you save it
- Guardrails throughout: real evidence always beats designation, closed periods are never touched, and each rule shows exactly what happened to every transaction its addresses cover
- Turning a rule off reverses it cleanly, and the tracking ledgers remain as the record of what it did

## Lot Selection & the Rules Engine

- Standard strategies: FIFO, LIFO, HIFO, LOFO, plus long-term-first and loss-first variants
  - Profile-scoped, priority-ordered rules that decide which strategy applies to each disposition type (sell, send, transfer, fee, option encumbrance)
  - Rules can target ledger attributes (e.g., treasury vs. exchange wallets)
  - Specific-identification (SpecID) overrides for hand-picking exact lots on any disposition
  - Lot-selection preview: see the tax result of each strategy before committing
  - Lot locks: fence off lots so automated selection never consumes them
  - Every allocation records *\*why\** that lot was chosen (strategy, rule, override) as a permanent, auditable reason code
- 

## PILLAR 02

# Ingestion, Matching, Pricing, and Data Integrity

Every path into Bedrock resolves into the same lot engine, lineage model, review workflow, and audit trail.

## Data Ingestion & Connectivity

Three paths in — and once data lands, the tax treatment, lineage, and audit trail are identical no matter which path it took.

- Direct API feeds to exchanges and custodians, with new venues added on customer demand
- On-chain wallet ingestion by address or xpub, with chain-agnostic downstream treatment
- Structured file import for OTC, private, and other activity that no API reports
- Identical tax treatment, lineage, and audit trail regardless of whether data arrived by feed, wallet, or file
- Opening-balance and opening-lot import for onboarding mid-history
- Incremental sync so new activity imports without rebuilding history
- Controlled onboarding and source management, including through-date imports, credential diagnostics, run reports, ledger tagging, and pause, resume, rewind, delete, and restore controls
- Nothing is silently dropped: records that fail to ingest are held for review, with a redacted row viewer, per-record acceptance, and one-click re-sync of the affected window
- Spam and dust token suppression at ingest, with a re-scan sweep for history already imported
- Fee-aware file import: fees denominated in any token or in fiat are priced at ingest and folded into basis and proceeds

## Transfers, Matching & Data Integrity

- Automatic transfer matching between your own wallets: internal moves are never mistaken for taxable sales
- Confidence-tiered matching (transaction hash > address > amount) with a recorded match score and reason
- Basis carries across matched transfers, split lot-by-lot from the source side
- Unmatched outflows surface as review items rather than silently becoming dispositions
- Canonical asset identity: ledgers and prices resolve by asset identity, not ticker, so same-symbol tokens from different contracts stay distinct end to end
- Duplicate and discrepancy detection with guided resolution
- Trade discrepancy review with side-by-side source comparison
- Validator and staking reward attribution by validator

## Pricing & Valuation

- Source-provided prices used first: the price your exchange reported is the price Bedrock books
- Independent market-price oracle as fallback
- Prices resolved by asset identity, so two tokens sharing a ticker never inherit each other's marks
- Historical price cache for consistent, reproducible valuations
- Manual price override on fair-value computations, with audit trail
- Trade-discrepancy price selection when sources disagree

---

## PILLAR 03

# Activity Coverage and Classification

Bedrock classifies protocol activity, user decisions, tags, notes, and ledger organization in the same governed record.

## DeFi & Protocol Activity

- Lending and borrowing: loan proceeds and repayments classified distinctly, with pledged collateral tracked without double-counting it in reconciliation
- Liquidity positions valued from the assets backing them rather than a quoted price on the receipt token
- Bridge crossings recognized as your own transfers between chains, not as sales
- Token migrations and contract upgrades carry cost basis and holding period across the swap
- Protocol settlement activity classified in its own right
- Gas and protocol fees classified and allocated under the same fee rules as everything else
- NFT detail: token standard and item identity carried through to transaction detail

## Categorization & Organization

- Activity categorization for income, staking, mining, payments, transfers, lending, bridging, and other source-attested events
- Three-tier category precedence: user override > rule > source-attested, so users retain final control
- Category override and revert per transaction, with audit trail
- Address book: named addresses with labels that appear across the timeline and filter views, with CSV import and export
- Organization and profile-level tags, with a rules engine to auto-apply tags by criteria and re-apply on demand
- Organization-defined custom fields and transaction notes
- Ledger display names, favorites, and archiving

---

## PILLAR 04

# Reconciliation, Transparency, and Explainability

Bedrock is designed to answer the review question behind every reported number: where did this come from?

## Reconciliation & Review

- Balance reconciliation: computed balances checked against custodian-reported balances, continuously
- Double-entry reconciliation report: every value movement counted exactly once; portfolio difference must collapse to zero
- Action Required: a single queue for everything needing a human decision: inventory shortfalls, over-encumbered lots, balance breaks, GAAP shortfalls, held-back records, unallocated fees, and external-transfer exceptions
- Outstanding items surface in navigation counts and in import completion emails, so nothing waits unseen
- Shortfall resolution workflows: acknowledge, escalate, or resolve with a synthetic deposit: each choice recorded
- Opening-balance roll-forward reconciliation for mid-history onboarding
- Options settlement reconciliation against underlying movements
- Escalation path to support built into every reconciliation surface

## Transparency — "Where Did This Number Come From?"

- Inline basis math on every lot and disposition: drill from a report number to the exact lots behind it
- Full lot lineage: parent/child links across splits and transfers, back to original acquisition
- Per-transaction running balances
- Transaction timeline view with export
- Source-record retrieval: view the raw record from the exchange or chain that produced a transaction
- Selection-reason on every allocation: the audit answer to "why this lot?"
- Related-transaction linking (fills, fees, settlements grouped together)
- Deterministic recalculation: identical inputs always produce identical outputs

---

## PILLAR 05

# GAAP Books, Fair Value, Reports, and Exports

The same activity can support tax books, GAAP books, reporting packages, and export workflows without rebuilding the record each time.

## GAAP Books & Fair Value

- Dual-book accounting: tax books and GAAP books maintained side by side from the same activity
- ASU 2023-08 fair-value measurement for crypto assets, computed per period
- GAAP asset-scope classification (which holdings are in scope for fair-value treatment)
- GAAP pools with effective-dated membership and average-cost or FIFO tracking
- Restricted holdings: positions under lockup carried as restricted, not freely disposable
- Event-sourced GAAP entries ready for journal-entry export
- GAAP shortfall detection and guided resolution
- Fair-value backing detail export

## Reports & Exports

- Report Center: self-service report builder with a growing catalog
- Tax and activity reports, including gain/loss, unrealized gains as of any date, token activity, and crypto income reporting
- Reconciliation balance sheet report
- Staking analysis report: validator positions, controlled withdraw authorities, and lockup exposure
- GAAP reports, including position, gain/loss, crypto income, and book/tax differences
- Excel downloads with full on-screen views and automatic bundling for very large outputs
- Filtered exports for audit trail, opening balance, timeline, transactions, lots, and audit views

## Controls, Access, Administration, and Platform Support

Bedrock treats controls, tenancy, help, and operations as part of the product surface, not as afterthoughts.

### Compliance, Audit & Controls (SOC 1)

- Designed for SOC 1 Type 2 from day one: controls, not promises
- Complete audit trail: every change records who, what, when, from where
- Tamper-evident audit log: database-level protection; records can never be edited or deleted, even by administrators
- Authorization denials are themselves audited events
- Four-eyes (maker-checker) approvals on sensitive operations: no one can approve their own proposal
- Per-organization compliance matrix: you decide which operations need approval, and how many approvers
- Period close and reopen with recorded snapshots
- Idempotent mutations: retries and double-clicks can never double-post
- Evidence-grade data model: lots, proposals, and approvals are never destructively deleted
- Reproducible calculations: decimal-exact math, no floating point, pinned regression scenarios

### Access, Tenancy & Administration

- Enterprise single sign-on with short-lived signed tokens
- Role-based access control over every operation in the product
- Fine-grained permission resolution drives the UI so users only see actions they are allowed to take
- Strict profile-level tenancy isolation, enforced at the data layer and verified at build time
- Organization administration and onboarding, including user invites, role grants and revocations, session revocation, profile switching, sandbox profiles, and per-feature elections
- Feature toggles that respect data safety, so a feature holding live data cannot be switched off silently

### Help, Support & Platform

- Bedrock Course: built-in guided learning track for the product and its tax concepts
- Inline contextual help on complex concepts with deeper documentation links
- In-product support escalation routed to a staff queue with full context attached
- Notifications, live status updates, and scheduled background processing for imports and recalculations
- Fully documented REST API through OpenAPI and Swagger: everything the UI does, customer systems can do through the API
- Calculation work is serialized per profile, so concurrent edits and imports can never interleave into an inconsistent result
- Architected for institutional volume: history grows, recalculation stays incremental, and reads stay paged