

Technical Integration

The Calyx Protocol is built to be open, auditable, and easy to integrate. Developers, verifiers, and organizations can connect via standard APIs, SDKs, and on-chain registries to mint, query, and propagate **BUDS (Blockchain Units of Derived Signal)**.

1. Architecture Overview

- **Application Layer** – Web dashboards and verifier-first tools for onboarding, verification submission, and proof consultation.
- **API Layer** – REST/GraphQL endpoints for integration into existing enterprise systems, sustainability platforms, and reporting tools.
- **On-Chain Layer** – Smart contracts anchor BUDS issuance, transfers, and propagation events immutably.
- **Registry Layer** – The Inter-Entity Attribution Ledger tracks lineage and prevents double counting of proof.

2. API Endpoints

All integrations begin at the API layer. Core endpoints include:

- **/join** – Register as a verifier (credentials and KYC checks).
- **/verify** – Submit a verification event (e.g., entity, metric, value, date).
- **/mint** – Trigger BUDS issuance for a verified fact.
- **/propagate** – Record proof propagation across entities.
- **/score** – Retrieve the current BUDS balance or compounded signal for an entity.
- **/ledger** – Query full lineage of an entity's BUDS across time and counterparties.

All responses include **transaction hashes** for on-chain verification and **audit trails** for compliance.

3. SDKs & Libraries

To simplify adoption, SDKs are provided in:

- **JavaScript/TypeScript** – for web platforms and dashboards.
- **Python** – for data science, sustainability reporting, and analytics pipelines.
- **Java** – for enterprise integrations with ERP and financial systems.

Each SDK includes pre-built functions for verification submission, BUDS balance checks, and ledger queries.

4. On-Chain Integration

- **Blockchain Choice** – Deployed on a scalable EVM-compatible chain with zk-readiness for privacy-preserving proofs.
- **Smart Contracts** – Govern issuance, transfer, and compounding of BUDS.
- **Anchoring** – Each verification event is hashed and recorded on-chain, ensuring immutability.
- **Attestation Layer** – Supports off-chain verification details (documents, certificates) anchored via IPFS or equivalent decentralized storage.

5. Verifier Dashboards

- **Submission Portal** – Upload and confirm verification events with metadata.
- **Audit View** – Inspect previously minted BUDS, proofs, and lineage.
- **Compounding View** – Visualize intra- and inter-entity alignment and resulting compounded issuance.
- **Export Tools** – API hooks to export verified proof into sustainability frameworks (GRI, SASB, CSRD) or financial reports.

6. Security & Authentication

- **Verifier Credentials** – Each verifier receives unique credentials (OAuth2 + JWT tokens).
- **Audit Logs** – Every action (submission, minting, transfer) is logged with timestamp, user ID, and transaction hash.
- **Data Integrity** – Hashes of documents and raw data are stored to prevent tampering.

7. Example Integration Flow

1. Verifier submits an emissions reduction event via /verify.
2. Calyx API validates credentials, formats data, and records metadata.
3. /mint endpoint mints **BUDS** tokens, anchored on-chain.
4. Entity's dashboard updates with new BUDS balance.
5. If another entity aligns (via /propagate), compounded issuance is triggered automatically.