

Institutional wallet requirements for Canton Network

This document outlines the set of capabilities we expect any institutional wallet provider to support on the Canton Network. Providers should treat this list as the minimum surface area needed to serve institutional clients on Canton, and reach out to us on anything unclear or that appears to conflict with an existing implementation.

■ **Required** ■ **Recommended**

Feature	Requirement status	Description
Canton Coin (CC)		Support the native utility token of the Global Synchronizer
Holding	■ Required	Allow users to hold CC and view their balance and transaction history
Transfer offers	■ Required	Allow users to initiate outgoing CC transfers and review, accept, or reject incoming transfer offers
Pre-approvals	■ Required	Allow users to create and revoke CC transfer pre-approvals, so senders can transfer CC without a separate recipient acceptance step
CIP-0056 Token Standard		Support assets issued through the Canton Network Token Standard
Holding	■ Required	Allow users to view their holdings and balances for CIP-0056-compatible assets
Transfers (/Transfer Offers)	■ Required	Allow users to initiate and track transfers of CIP-0056-compatible assets and review, accept, or reject incoming transfer offers
Allocations / DvP settlement	■ Required	Allow users to allocate assets to settlement workflows that support atomic delivery-versus-payment transactions in response to CIP-0056 Allocation Requests.
CIP-0112 Token Standard v2		Support assets issued through v2 of the Canton Network Token Standard. V2 support is additive, CIP-0056 support remains required and is not superseded
Accounts <code>splice-api-token-holding-v2</code>	■ Recommended	Represent holders, senders, and receivers as <code>Account</code> (owner / provider / id) rather than bare parties. Show or hide account input fields per the asset's <code>showAccountInputFields</code> flag, and present the <code>cip-112/mint</code> and <code>cip-112/burn</code> accounts as mints and burns rather than transfers
Holding <code>splice-api-token-holding-v2</code>	■ Required	Allow users to view holdings and balances grouped by account, and fund transfers only from holdings of the account being debited
Transfers <code>splice-api-token-transfer-instruction-v2</code>	■ Required	Allow users to initiate and track transfers, driving the available operations from <code>availableActions</code> (which replaces the V1 <code>status</code> field) rather than from inferred state. Allow users to review, accept, or reject incoming transfer offers.
Allocations / DvP settlement <code>splice-api-token-allocation-*v2</code>	■ Required	Allow users to authorize a settlement by creating one allocation per <code>AllocationSpecification</code> , in a single signature, after displaying the complete <code>SettlementInfo</code> including metadata and every transfer leg
Committed & iterated allocations <code>splice-api-token-allocation-allocation-v2</code>	■ Required	Allow users to create allocations that are committed (not unilaterally withdrawable) or that reserve funding for repeated settlement, and make clear which of the two they are signing
Transaction history <code>splice-api-token-transfer-events-v2</code>	■ Recommended	Derive holding and transfer history from <code>EventLog_HoldingsChange</code> events for v2 assets, retaining V1 factory-choice parsing for v1-only assets

Version selection (supportedApis)	■ Recommended	Select the interface version per asset and per request; call the V1 factory when the asset or the request is V1-only, and warn the user when a request asks for features the asset cannot honor
Batching	■ Recommended	Combine multiple token standard actions into a single transaction using BatchingUtilityV2
CIP-0103 dApp Connectivity		Allow users to securely connect their wallet to Canton Network applications
CIP-0103 dApp API	■ Required	Implement the CIP-0103 dApp API (sync or async) for transaction signing/submission, events, relayed ledger reads, and errors
CIP-0103 via Wallet Connect	■ Recommended	Comply with CIP-0103 dApp API through WalletConnect, which allows users to establish sessions using remote web or mobile wallets via tunneling
Validator Setup		Offer a broad set of validator deployment models so customers can choose the setup that best matches their infrastructure, security, privacy, and operational preferences. Wallet providers can support one, some, or all validator setups based on customer needs. Over time, we recommend supporting all three, as customer requirements often change late in the implementation process.
Bring Your Own Validator (BYOV)	■ Recommended	Allow an institution to connect the wallet to a validator it operates or selects, without requiring the wallet provider to host the validator
Private validator	■ Recommended	Support a dedicated validator environment for a single institution, with isolated infrastructure, access controls, data, and operational responsibilities
Shared validator	■ Recommended	Support a provider-operated validator environment shared across multiple customers
Other		
Memo tag support for transfers to omnibus accounts	■ Required	Allow users to enter, validate, display, and preserve a memo or destination tag so deposits sent to a shared omnibus party can be attributed to the correct underlying customer.
UTXO merge delegation contract enabled	■ Required	Allow the user to authorize a trusted operator or service to merge fragmented token holdings on their behalf, reducing UTXO fragmentation without granting unrestricted transfer authority. Offer UTXO merge automation to customers
Clear signing	■ Recommended	Present the human-readable intent of a (prepared) transaction before approval, rather than asking users to approve an opaque hash
Wallet Gateway signing driver	■ Required	Provide a Wallet Gateway signing driver that receives prepared transactions, applies institutional authorization policies, obtains signatures from the custody system, and returns them for submission
Digital Asset Token Registry assets		Support the Digital Asset operated token registry workflows and assets required for institutional use.
Pre-approvals	■ Required	Allow users to discover, create, review, and revoke registry-specific transfer pre-approvals for supported DA Registry assets
USDCx	■ Required	Support holding, balance visibility, deposits, withdrawals, and Token Standard transfer workflows for USDCx