NDD Circulation Beyond Deposit Accounts: Non-NDDID Holders, Segregated Lockboxes, and Compliance Gates

May 2026

Aurélien Bonnel
aurelien.bonnel@n3xt.io

Elise LaPunzina
elise.lapunzina@n3xt.io

Kyle O'Donnell
kyle.odonnell@n3xt.io

Tiffiney Peterson
tiffiney.peterson@n3xt.io

Jeffrey Wallis
jeffrey.wallis@n3xt.io

Scott Shay
scott.shay@n3xt.io

Abstract

The Tokenized Deposit (N3XT Digital Dollar, NDD) described in our base architecture is initially designed as a closed-loop instrument: only wallets holding a non-transferable Identity NFT (NDDID) may custody and transfer tokens, enabling strong on-chain compliance. This companion paper specifies an **extension** in which NDD may also circulate in wallets held by **non-NDDID holders**—addresses without an NDDID that are explicitly admitted via an on-chain allow list—while preserving full-reserve backing, segregated reserve accounting, and the exclusive right of **NDDID holders** to **request** mint, USD redemption, and reserve-releasing burn—operations that only N3XT may execute as sole minter, redeemer, and burner on-chain. We define the **Non-NDDID Lockbox** and **Frozen Lockbox** on the Accounting Ledger, mandatory **journal entries in the ledger of accounts** for all lockbox boundary crossings, administrative **freeze/unfreeze** for policy- and compliance-driven holds, and an **on-chain allow list** on the NDDIDv2 identity contract that gates non-NDDID-holder participation. The extension is designed to layer on the real-time reconciliation, Digital Lockbox, and Core Ledger model of the base specification without weakening its reserve guarantees.

1 Relationship to the Base Specification

This document is a **companion whitepaper**. The foundational treatment of the Tokenized Deposit—including the SPDI full-reserve model, the three-layer architecture (blockchain, Accounting Ledger, reconciliation infrastructure), the Digital Lockbox, mint and redemption flows, peer-to-peer reconciliation, proof-of-reserves invariants, and the Identity NFT (NDDID) compliance framework—is given in the **Base Specification** [1], *The Tokenized Deposit: Real-Time Settlement Infrastructure for the Digital Economy*.

This paper does not restate the full reconciliation pipeline (Block-Listener, Event Broker, Reconciliation Service, Minter/Burner); it specifies *only* the additional elements related to NDD circulation outside of the NDDID-only closed loop. Where terminology overlaps (e.g., Digital Lockbox, `totalSupply()`, Core Ledger, Accounting Ledger), meanings align with the Base Specification unless explicitly extended below.

2 Roles and Transfer Permissions

- **NDDID holder:** A legal entity onboarded to the bank with a KYB/KYC/BSA-compliant deposit relationship. Their vetted wallet(s) hold an NDDID (Identity NFT minted via onboarding). Only NDDID holders may **request** a **mint** to create NDD against fiat on deposit, a **redemption** for USD payout, or a **burn** as part of an institution-approved withdrawal or recovery flow; **N3XT** is the sole minter, redeemer, and burner on-chain and signs the corresponding transactions. NDDID holders are **always eligible** to send and receive NDD: no allow-list entry is required.

- **Allow-listed non-NDDID holder:** A wallet that does *not* hold an NDDID but ap-

pears on the on-chain allow list maintained in **NDDIDv2**. Such wallets may **send and receive** NDD subject to `validateIdentity` / `validateTransferIdentities`. They may **not request** mint, redemption for USD, or reserve-releasing burn; those operations are available only to NDDID holders and are executed by N3XT.

- **Other:** Any address without an NDDID and not on the allow list **cannot** participate in NDD transfers under this extension.

Transfer eligibility. For any NDD transfer, both `from` and `to` must satisfy identity validation on NDDIDv2—either **(1)** the address holds an NDDID NFT, or **(2)** the address is on the on-chain allow list. The contract enforces this via `validateIdentity(addr)` and `validateTransferIdentities(from, to)`. There is **no separate on-chain deny registry**; ineligible addresses simply fail validation.

Permitted token movements (extension mode). Each leg requires both counterparties to be eligible as above.

- NDDID holder wallet → NDDID holder wallet (unchanged from the Base Specification; compliance enforced via NDDID rules on-chain).
- NDDID holder wallet → allow-listed non-NDDID holder wallet.
- Allow-listed non-NDDID holder wallet → allow-listed non-NDDID holder wallet.
- Allow-listed non-NDDID holder wallet → NDDID holder wallet (return of value to the vetted ecosystem).

Figures 1–4 depict end-to-end sequence flows for the four permitted movements in Section 2. Before settlement, the token contract consults NDDIDv2: `validateTransferIdentities(from, to)` requires each party to hold an NDDID NFT or appear on the on-chain allow list. Paths involving a non-NDDID holder additionally trigger reserve reclassification in the Reconciliation Service as shown.

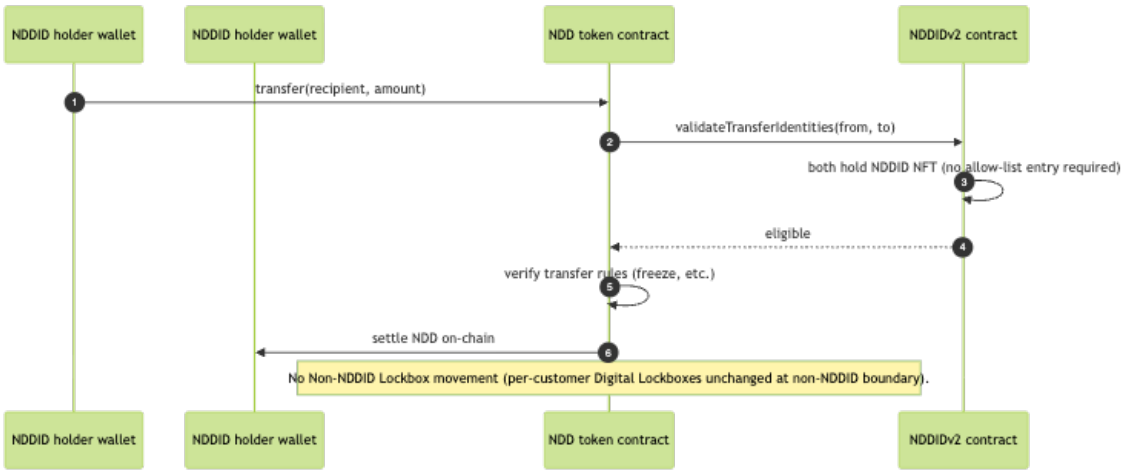

Figure 1: NDD transfer sequence: NDDID holder to NDDID holder (closed loop). NDDIDv2 identity checks per Base Specification; no allow-list entry required.

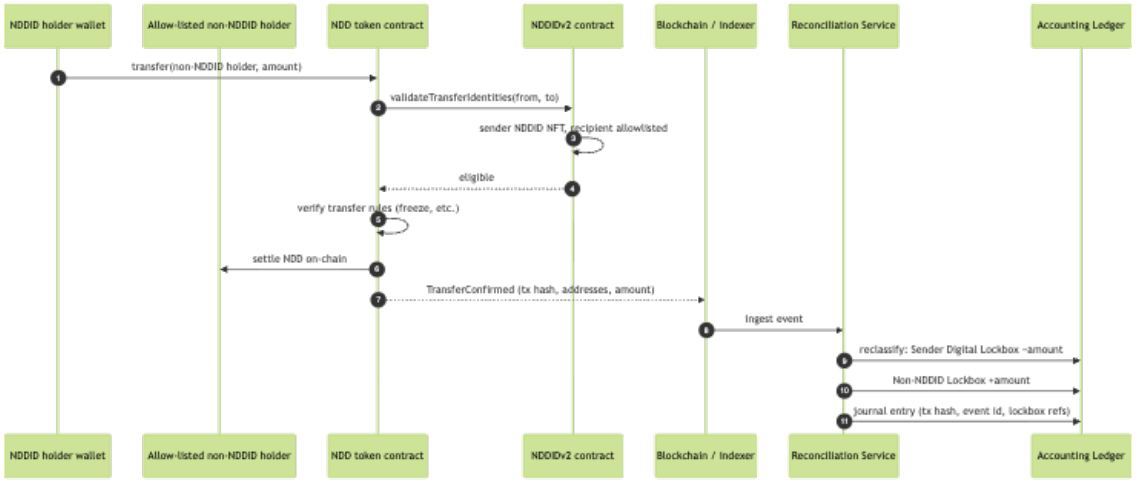

Figure 2: NDD transfer sequence: NDDID holder to allow-listed non-NDDID holder. NDDIDv2 identity check; Reconciliation Service reclassifies reserves from the sender Digital Lockbox to the Non-NDDID Lockbox with journal entries.

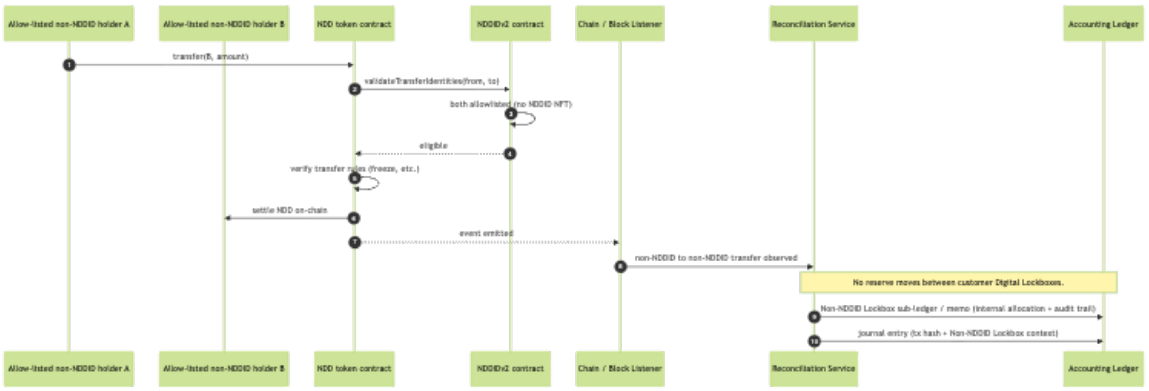

Figure 3: NDD transfer sequence: allow-listed non-NDDID holder to allow-listed non-NDDID holder. NDDIDv2 identity check; no cross–Digital Lockbox reserve moves between customers—only Non-NDDID Lockbox aggregate and sub-ledger updates.

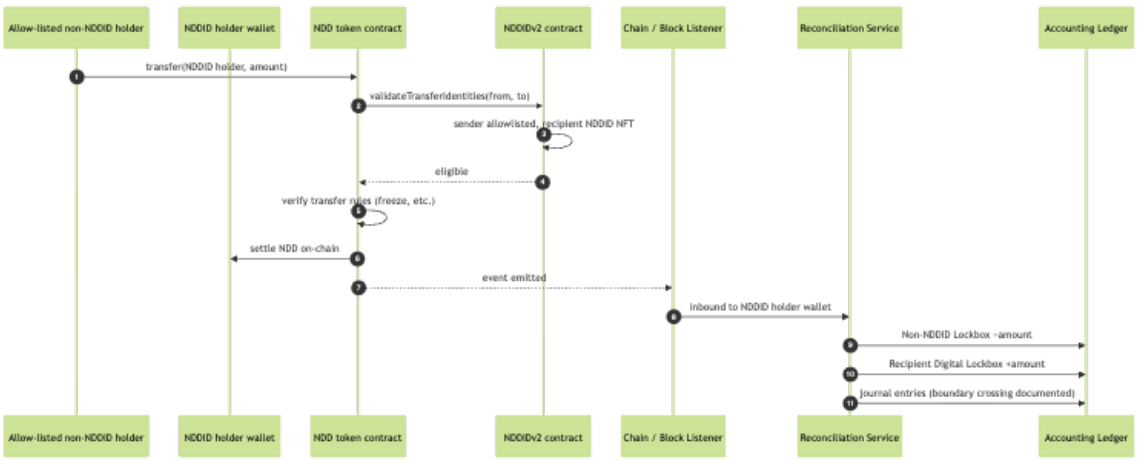

Figure 4: NDD transfer sequence: allow-listed non-NDDID holder to NDDID holder. NDDIDv2 identity check; reserves return from the Non-NDDID Lockbox to the recipient Digital Lockbox.

3 Segregated Reserves: Non-NDDID Lockbox and Frozen Lockbox

In the Base Specification, each NDDID holder's tokenized balance is backed by reserves in a customer-specific **Digital Lockbox**. Under this extension, *all* reserves economically backing tokens held by non-NDDID holders are aggregated in a single institution-controlled **Non-NDDID Lockbox** on the Accounting Ledger. N3XT is the custodian of record for this lockbox: it is not commingled with any single client's Digital Lockbox balance, and it exists precisely to mirror the liability represented by NDD held by non-NDDID holders on-chain.

Reconciliation mapping. When NDD moves from an NDDID holder wallet to an allow-listed non-NDDID holder wallet, the Reconciliation Service does *not* destroy the sender's reserve backing; it **reclassifies** the corresponding dollar amount from the sender's Digital Lockbox into the Non-NDDID Lockbox, with a full audit trail (blockchain tx hash, from/to addresses, amount). Transfers between two allow-listed non-NDDID holders reallocate **no** reserve between client Digital Lockboxes—only off-chain sub-ledger entries within the Non-NDDID Lockbox aggregate (and on-chain balances) move. When NDD returns from an allow-listed non-NDDID holder wallet to an NDDID holder wallet, reserves move from the Non-NDDID Lockbox back into the recipient's Digital Lockbox.

Invariant (extension). Full reserve backing is preserved by extending the verification identity:

$$\texttt{totalSupply()} = \sum_i \text{DigitalLockbox}_i + \text{Non-NDDID Lockbox} + \text{Frozen Lockbox} \tag{1}$$

where the **Frozen Lockbox** holds the dollar equivalent of on-chain balances that have been administratively frozen (Section 5). Operational verification sums all three (plus any explicit transit/pending buckets the institution uses) against `totalSupply()`.

4 Ledger of Accounts: Journal Entries

Every movement of funds **into** or **out of** the Non-NDDID Lockbox (including reclassification from a customer's Digital Lockbox on outbound transfer to a non-NDDID holder, and the reverse on inbound transfer from a non-NDDID holder to an NDDID holder wallet) must generate a corresponding **journal entry in the ledger of accounts**, with references to the blockchain transaction hash, internal event ID, and lockbox sub-account. This ensures that non-NDDID-backed liabilities are visible in the same financial reporting fabric as traditional deposits and Digital Lockboxes, satisfying audit, regulatory examination, and management reporting without manual shadow spreadsheets.

Similarly, any movement between the Non-NDDID Lockbox and the **Frozen Lockbox** on freeze or unfreeze must be **posted to the accounting ledger** with explicit memo codes (e.g., internal case reference, operations ticket identifier, or policy memo code) to support audit and forensic reconstruction.

5 Administrative Freeze and Unfreeze

The institution requires an on-chain and ledger-aligned **freeze** capability for regulatory and legal purposes.

Behavior. When a wallet is frozen, its NDD balance remains on-chain but the token contract (or a mandatory transfer hook) **blocks all transfers** except pre-defined administrative paths (e.g., transfer to an institution-controlled settlement wallet or burn executed only through authorized custodian workflows). For allow-listed non-NDDID-holder wallets, freeze is applied in a defined order: **(1)** remove the address from the on-chain allow list on NDDIDv2; **(2)** move the dollar equivalent of the frozen on-chain balance in the Accounting Ledger from the Non-NDDID Lockbox into the **Frozen Lockbox** (and, on the token side, move balances to the frozen lockbox per NUSD contract policy). The Frozen Lockbox is a **reserve and liability sub-ledger bucket** on the Accounting Ledger—not an on-chain sweep of tokens into a separate vault; NDD stays in the frozen address until an administrative path moves or destroys it. For NDDID holders, the frozen balance is still visible in their Digital Lockbox but requests for mint or redemption will fail and the balance is untransferable. This separation is **accounting-driven**: regulators and auditors can see restricted funds distinct from non-NDDID-backed reserves.

Unfreeze. When the hold is released, the **unfreeze** reverses the sequence: **(1)** move reserves from the Frozen Lockbox back to the wallet's appropriate lockbox (Non-NDDID Lockbox or customer Digital Lockbox) on the NUSD / Accounting Ledger side; **(2)** restore the address to the on-chain allow list on NDDIDv2 where applicable; **(3)** restore transfer rights on-chain, with **journal entries in the ledger of accounts** mirroring each step.

6 On-Chain Allow List (NDDIDv2)

Transfers between NDDID holders continue to rely on on-chain NDDID checks as in the Base Specification. For any transfer where **either** party is not an NDDID holder, participation requires membership on an **on-chain allow list** stored in **NDDIDv2.sol**.

Contract integration. NDDIDv2 maintains an `allowlisted` mapping and exposes `validateIdentity(addr)`, which returns true if the address holds an NDDID NFT **or** is allowlisted. `validateTransferIdentities(from, to)` requires **both** endpoints to pass. The NDD token contract (or transfer hook) invokes these checks before settlement. Privileged roles (`ALLOWLIST_ADMIN` or institution automation) may call `addAllowlist(addr)` and `removeAllowlist(addr)`. There is no separate on-chain deny registry: addresses not on the list and without an NFT simply cannot transfer.

Off-chain admission. Screening, terms acceptance, and case management occur before an address is added. Wallet screening (e.g., TRM) and compliance review gate whether a submitted address is persisted and subsequently pushed on-chain by an **Updater** component; removals follow rescreen failures or manual compliance action (Section 7).

7 Allow-List Operations and List Maintenance

Request flow. A customer submits a wallet address via a signed allow-list request on the N3XT web property. The API verifies the signature, creates a request record, and triggers wallet screening. On pass, the wallet is persisted; an Updater (operations or automation) submits `addAllowlist` to NDDIDv2. On fail, the request is rejected with reason. Re-screening may trigger automated `removeAllowlist`.

Keeping the on-chain set current. At launch, the institution may combine **manual** incremental add/remove transactions (moderate latency, low engineering effort) with **automated adds** from screening passes (e.g., webhook-driven submission on TRM approval). Automated **removals** on rescreen failure are recommended in parallel. A periodic compliance-database snapshot diffed against the on-chain set (batch `add[]` / `remove[]`) is a moderate-effort alternative for scheduled reconciliation. If the allow set grows large, a **Merkle allow root** on NDDIDv2 with incremental diffs, or oracle/automation feeds updating the root on cadence, may be justified—these are deferred until scale warrants the added contract and operations complexity.

Counterparty visibility. Because both parties must be allow-listed or NDDID holders before a transfer succeeds, a sender cannot infer from the chain alone whether an intended recipient has completed onboarding. Product and API surfaces may expose eligibility or request status to signed-in users; whether counterparty pre-check is required for acceptable UX remains an open operational question.

8 NDDID and the On-Chain Allow List: A Proactive On-Chain Compliance Stack

Fiat-backed tokens with centralized issuance have become the dominant form of on-chain dollar liquidity. Their compliance model is well documented in industry practice and in security research: issuers maintain smart-contract mechanisms to block transfers from or to flagged addresses, and may freeze balances *after* a wallet is identified as high-risk. That pattern is **reactive at the address level**: any address can typically receive tokens by default; enforcement applies once the issuer updates on-chain state. Because a deny/freeze invocation is itself an ordinary transaction on a public network, it can in principle be ordered *after* a competing transfer in the same block—so the *effectiveness* of a freeze can depend on mempool dynamics and block production economics, not only on issuer policy [2].

8.1 From reactive lists to layered gates

The NDD extension combines two complementary mechanisms that operate *before* value settles on the non-NDDID-holder surface:

1. **NDDID (non-transferable Identity NFT):** Establishes a **vetted admission** layer for NDDID holders. Only onboarded NDDID holders can request mint, redemption, or reserve-releasing burn (executed by N3XT) and bind reserves in their Digital Lockboxes; the closed loop is the proactive baseline.

2. **On-chain allow list (NDDIDv2):** For every transfer involving a non-NDDID holder, the contract requires **both** endpoints to be either NDDID-verified or explicitly allow-listed *prior* to transfer execution. Admission is proactive—wallets are screened and approved before they appear on-chain—rather than open to all addresses until individually denied.

Together, these layers shift a material portion of compliance from "flag an address after the fact" to "**prove eligibility before each settlement involving a non-NDDID holder**." Policy changes take effect when the allow list is updated on NDDIDv2 (and, for frozen wallets, when allow-list removal and lockbox moves complete). Emergency removal from the allow list can halt circulation for a non-NDDID-holder address without redeploying core token logic.

8.2 Comparison to the conventional fiat-backed token compliance pattern

Table 1 summarizes the positioning at a high level. The left column reflects the *prevailing* issuer-controlled design described in public documentation and in blockchain security literature; the right column reflects the NDD extension as specified in this paper. Neither column captures every product variant; the goal is to make the architectural difference explicit.

Conventional fiat-backed token (issuer-controlled)	**NDD: NDDID + on-chain allow list (extension)**
Default holder set: Any address can usually receive and self-custody tokens until individually denied or frozen.	**Default holder set:** NDDID holders are NDDID-gated; non-NDDID holders can hold and move NDD only after explicit allow-list admission, not as ungated universal bearer dollars.
Open-loop transfers: Peer transfers are generally allowed between arbitrary addresses subject to post-hoc contract checks against a static mapping.	**Open-loop transfers:** NDDID-holder↔NDDID-holder transfers use the Identity NFT framework; any transfer involving a non-NDDID holder requires both parties to pass `validateTransferIdentities` on NDDIDv2.
Policy refresh: Denylist/freeze updates apply when the issuer's on-chain transaction confirms; transfers may occur under stale policy between updates.	**Policy refresh:** Allow-list and freeze state on NDDIDv2 and the token contract apply at transfer time; off-chain screening and Updater cadence keep the on-chain set aligned with compliance systems of record.
Reserve accounting: Reserves are usually tracked at issuer omnibus level; not tied to per-customer Digital Lockboxes as in the Base Specification.	**Reserve accounting:** Non-NDDID Lockbox + Frozen Lockbox + per-customer Digital Lockboxes reconcile to `totalSupply()` (Section 3).
Enforcement timing: Contract authority is necessary but, on public networks, not always sufficient for instantaneous freezes relative to same-block transfers [2].	**Enforcement timing:** On-chain identity gating does not remove consensus-layer ordering economics, but it reduces the non-NDDID-holder attack surface by refusing *ineligible* transfers *before* they settle; NDDID-holder-only flows remain a separate, pre-vetted lane.

Table 1: High-level compliance architecture: conventional issuer-controlled fiat-backed tokens versus the NDD extension. Not exhaustive of all market implementations.

9 Summary of Compliance and Risk Posture

This extension broadens **who may hold and move** NDD on public chains without broadening

who may request mint, redemption for USD, or reserve-releasing burn—each of which remains initiated by NDDID holders and executed by N3XT as sole minter, redeemer, and burner. Reserves for non-NDDID-holder circulation are always in a **segregated Non-NDDID Lockbox**; frozen funds are in a **Frozen Lockbox**; the **accounting ledger** records every lockbox boundary crossing; and the **on-chain allow list on NDDIDv2** enforces that both transfer parties are NDDID holders or are explicitly admitted before any settlement involving a non-NDDID holder. Together, these controls preserve the full-reserve and real-time reconciliation guarantees of the Base Specification while enabling controlled openness to the wider wallet ecosystem.

References

[1] N3XT. (2025). *The Tokenized Deposit: Real-Time Settlement Infrastructure for the Digital Economy*. Companion technical whitepaper. https://static-assets.n3xt.io/NDD.pdf

[2] Wu, D., Bai, Y., Ren, S., Zhang, X., Cao, Y., Wang, X., Wen, W., & Liu, J. (2026). *Ordering Power is Sanctioning Power: Sanction Evasion-MEV and the Limits of On-Chain Enforcement*. arXiv:2603.27739. https://arxiv.org/abs/2603.27739