

DEMt Bridge — Whitepaper

A DEM ■ TRON tokenization bridge (Testnet: Nile)

Version 1.0 • November 02, 2025

Overview

DEMt Bridge links the classic Deutsche eMark (DEM) blockchain to the TRON network by issuing a tokenized representation of DEM on TRON named DEMt. Each DEMt is intended to be backed 1:1 by a verifiable DEM reserve on the native chain. The project currently operates on the TRON Nile testnet for public testing and transparency.

Objectives

- Provide a secure, auditable 1:1 mapping between DEM and DEMt.
- Enable DEM utility in smart contracts, DApps, and DeFi on TRON.
- Keep the system open, observable, and operationally transparent.

Architecture

The bridge is composed of four core components:

- 1) Backend Server (Node.js + PostgreSQL): Orchestrates deposits, mints, and redeems; validates signatures and on-chain events; exposes REST/WebSocket APIs.
- 2) Agent Node: Monitors the DEM chain for inbound deposits; after sufficient confirmations, it triggers mint signatures on TRON.
- 3) DEM Infrastructure: A fully synchronized ElectrumX server and full nodes provide reliable access to balances, UTXOs, and transaction histories.
- 4) Frontend (vanilla HTML/JS, no framework): Connects via TronLink for authentication and enables users to create deposits, execute mints, and initiate redeems with live status updates.

Token Model

- Name: Deutsche eMark TRON
- Symbol: DEMt
- Decimals: 8
- Supply: Dynamic, minted on demand against DEM reserves and burned on redeem.
- Optional Fees: Configurable mint fee (e.g., 1%) to cover operational and on-chain costs.
- Use Cases: Payments, DeFi integrations, programmatic transfers.

Security

- Wallet auth via TronLink and short-lived JWTs.
- Deposit recognition only after a confirmation target (default: 30) on DEM.
- State machine with explicit transitions (READY_TO_MINT → MINTING → MINTED), watchdogs, and finalizers.
- Consistency checks: duplicate prevention, idempotent confirmations, and audited logs.

- Separation of roles for hot/cold operation where applicable.

Testnet & Status

- Network: TRON Nile (testnet) for integration and public testing.
- Backend: Modular Node/Express server, JWT auth, TronWeb v6 integration, Postgres persistence.
- Agent: WebSocket-secured bridge link (PI_SHARED_KEY) emitting DEPOSIT_* events and recomputing status.
- DEM Backend: ElectrumX-based indexer, full-node connectivity for UTXO-level validation.
- Frontend: Lightweight vanilla HTML/JS application; no frontend framework used.
- Current: Mint, redeem, and sweep flows functional; UI polishing ongoing.

Compliance & Transparency

- Public contract verification on TronScan is required before mainnet launch.
- Testnet contracts and code are intended for demonstration and audits only.
- Reserve attestations and operational policies will be published prior to mainnet activation.

Contact

- Website: swap-pool.de
- GitHub: github.com/djspacedevil/demt-bridge
- Email: info@swap-pool.de

DEMt Bridge — Whitepaper

Eine DEM ■ TRON Tokenisierungs-Bridge (Testnetz: Nile)

Version 1.0 • 02. November 2025

Übersicht

Die DEMt Bridge verbindet die klassische Deutsche-eMark-Blockchain (DEM) mit dem TRON-Netzwerk, indem eine tokenisierte Repräsentation von DEM auf TRON als DEMt ausgegeben wird. Jeder DEMt soll durch eine verifizierbare DEM-Reserve im Verhältnis 1:1 gedeckt sein. Das Projekt läuft derzeit auf dem TRON-Testnetz Nile zu Test- und Transparenz-zwecken.

Ziele

- Sichere, prüfbare 1:1-Abbildung zwischen DEM und DEMt.
- Nutzung von DEM in Smart-Contracts, DApps und DeFi auf TRON ermöglichen.
- Offene, beobachtbare und transparent betriebene Architektur.

Architektur

Die Bridge besteht aus vier Hauptkomponenten:

- 1) Backend-Server (Node.js + PostgreSQL): Orchestriert Deposits, Mints und Redeems; validiert Signaturen und On-Chain-Ereignisse; stellt REST/WebSocket-APIs bereit.
- 2) Agent-Node: Überwacht die DEM-Blockchain auf Eingänge; nach ausreichenden Bestätigungen werden Mint-Signaturen auf TRON ausgelöst.
- 3) DEM-Infrastruktur: Vollständig synchronisierte ElectrumX-Instanz und Full-Nodes liefern zuverlässige Balances, UTXOs und Transaktionshistorien.
- 4) Frontend (vanilla HTML/JS, ohne Framework): Verbindet sich über TronLink zur Authentifizierung und ermöglicht Deposits, Mints und Redeems mit Live-Status.

Token-Modell

- Name: Deutsche eMark TRON
- Symbol: DEMt
- Decimals: 8
- Supply: Dynamisch; Mint auf Nachfrage gegen DEM-Reserve, Burn bei Redeem.
- Optionale Gebühren: Konfigurierbare Mint-Fee (z. B. 1 %) zur Deckung von Betriebs- und On-Chain-Kosten.
- Einsatzzwecke: Zahlungen, DeFi-Integrationen, programmatische Transfers.

Sicherheit

- Wallet-Authentifizierung via TronLink und kurzlebigen JWTs.
- Deposit-Anerkennung erst nach Erreichen eines Bestätigungsziels (Standard: 30) auf DEM.
- Zustandsautomat mit klaren Transitionen (READY_TO_MINT → MINTING → MINTED), Watchdogs und Finalizern.

- Konsistenzprüfungen: Duplicate-Prävention, idempotente Bestätigungen, auditierbare Logs.
- Rollentrennung für Hot/Cold-Betrieb, wo sinnvoll.

Testnetz & Status

- Netzwerk: TRON Nile (Testnetz) für Integration und öffentliche Tests.
- Backend: Modularer Node/Express-Server, JWT-Auth, TronWeb v6, Postgres-Persistenz.
- Agent: WebSocket-gesicherte Bridge-Anbindung (PI_SHARED_KEY), sendet DEPOSIT_*-Events und berechnet Status neu.
- DEM-Backend: ElectrumX-basierter Indexer, Full-Node-Anbindung für UTXO-Validierung.
- Frontend: Schlanke Anwendung in Vanilla HTML/JS; es wird **kein** Frontend-Framework verwendet.
- Stand: Mint-, Redeem- und Sweep-Flows funktionsfähig; UI-Feinschliff läuft.

Compliance & Transparenz

- Öffentliche Contract-Verifizierung auf TronScan ist vor Mainnet-Start vorgesehen.
- Testnetz-Verträge und Code dienen Demonstration und Audits.
- Reserve-Atteste und Betriebsrichtlinien werden vor Mainnet-Aktivierung veröffentlicht.

Kontakt

- Website: swap-pool.de
- GitHub: github.com/djspacedevil/demt-bridge
- E-Mail: info@swap-pool.de