

Smart Contract Integration

Platform & Architecture · Published by UnyKorn LLC (Wyoming), technology provider · September 2026

DEMONSTRATION DOCUMENT — NOT AN OFFERING, NOT A BOUND CONTRACT. All entities, identifiers, reserves, serials, vaults, policies, counterparties and figures in this document are illustrative and synthetic. UnyKorn LLC is not a broker-dealer, exchange, alternative trading system, investment adviser, bank, trust company, qualified custodian, money transmitter, insurer or issuer, and does not hold client assets or funds. No entity described here is registered with the SEC, FINRA, the FCA or any other regulator. No refinery, vault operator, insurer, broker, auditor or qualified person named or implied is a counterparty to UnyKorn LLC. Nothing here is an offer to sell or a solicitation of an offer to buy any security, or investment, legal, tax or accounting advice.

DGX SMART CONTRACT INTEGRATION & ABI GUIDE

Technical Reference for Smart Contracts in `contracts/src/`

Contract Suite Overview

1. `DGXBullionToken.sol` : 1:1 Allocated Bullion Token (ERC-20 / ERC-3643 compliant)
2. `MiningStreamingForge.sol` : In-Ground Mining Reserve & Forward Streaming Contract
3. `AtomicSettlementDvP.sol` : Institutional Delivery-versus-Payment Escrow
4. `ProofOfReservesOracle.sol` : Multi-Sig Attested Depository Telemetry Feed
5. `IoTEsgOffsetRegistry.sol` : Mine Telemetry & Carbon Offset Retirement

Solidity Interfaces & ABI Signatures

DGXBullionToken Interface

```

interface IDGXBullionToken {
    function enrollBarPassport(
        string calldata serialNumber,
        string calldata refiner,
        uint16 finenessBps,
        uint256 fineWeightGrams,
        string calldata vaultEnclave,
        string calldata insurancePolicyHash
    ) external returns (bytes32);

    function mintBullion(address recipient, uint256 amount, bytes32 barHash) external;
    function redeemPhysical(uint256 amount, string calldata physicalDeliveryAddress) external;
    function setWhitelisted(address account, bool status) external;
    function balanceOf(address account) external view returns (uint256);
}

```

MiningStreamingForge Interface

```

interface IMiningStreamingForge {
    function enrollConcession(
        string calldata concessionId,
        string calldata mineName,
        string calldata jurisdiction,
        uint256 provenReservesOz,
        uint256 upfrontCapexUsd,
        uint256 fixedPricePerOzUsd,
        uint256 totalOzCommitted,
        string calldata technicalReportFilingHash
    ) external returns (bytes32);

    function logRefineryDelivery(
        bytes32 concessionHash,
        uint256 ozDelivered,
        string calldata refineryRef
    ) external;
}

```

AtomicSettlementDvP Interface

```
interface IAtomicSettlementDvP {  
    function createOrder(  
        address seller,  
        address paymentToken,  
        address assetToken,  
        uint256 paymentAmount,  
        uint256 assetAmount,  
        uint256 validityDurationSecs  
    ) external returns (uint256);  
  
    function executeAtomicSettlement(uint256 orderId) external;  
    function cancelOrder(uint256 orderId) external;  
}
```