

USDPROP — Smart Contract Structure and Deployment Report

This document details the smart contract architecture of the **USDPROP** protocol, specifying the contract types (standard, proxies, and implementations), their active addresses recorded after the complete deployment (`deployment.json`), their responsibilities within the system, and how they are wired to secure investment flows and compliance.

1. General Deployment Overview

The USDPROP protocol is composed of **12 active interconnected contracts**. To optimize gas consumption and allow secure upgrades to execution logic without altering the system's state or the contract addresses, the system utilizes the **UUPS Proxy (ERC-1967)** standard for its core components.

- **Active Access Addresses (Entrypoints):** 12
 - **Contracts with UUPS Proxy Pattern:** 6 (each composed of a Proxy contract and an Implementation contract).
 - **Standard / Single-Register Contracts:** 6
 - **Test Environments:** 1 additional contract (`MockUSDC`) is optionally deployed to simulate the stable investment currency.
-

2. Detailed Contract Inventory

The following table summarizes the contracts that compose the active topology registered in production (**Chain ID: 137** - Polygon Mainnet):

#	Contract Name	Active Address (Proxy/Direct)	Contract Type	Purpose / Responsibility
1	<code>ClaimTopicsRegistry</code>	<code>0x00D23AA72703B79fF708E5B1c98757074Be19077</code>	Standard	Stores the mandatory claim topics required by the system (e.g., KYC, Sanctions).
2	<code>TrustedIssuersRegistry</code>	<code>0x60F561C93106d4805FEcFe8980DEea0829115475</code>	Standard	Contains the list of registry addresses of valid claim issuer contracts.

#	Contract Name	Active Address (Proxy/Direct)	Contract Type	Purpose / Responsibility
3	ClaimIssuer	0x656B7DDFf86ce7aBE9E86A8bf97642d55B95a61a	Standard	On-chain identity of the KYC/AML service, responsible for digitally signing investor claims.
4	IdentityRegistryStorage	0xA017992Fddc1A65d6dD8b753E54cf5cA3E466201	Standard	Persistent database mapping investor wallet addresses to their corresponding ONCHAINIDs.
5	IdentityRegistry	0x82e04Ac5abC1c6979781aeB2ACa5133B59f86bac	Standard	Central module of the ERC-3643 standard that verifies the existence and validity of claims to authorize transfers.
6	ModularCompliance	0x42a133b07c53FEb1043Ac84481993ceec72D1294	UUPS Proxy	Compliance rules orchestrator that interacts directly with the token during each transfer.
7	USDPROPComplianceModule	0x6A7863dF05D06956bfE8B7650f2c2983e214bDcc	UUPS Proxy	Module containing regulatory business logic (Reg D/S locks, OFAC sanctions, country caps, and holding limits).

#	Contract Name	Active Address (Proxy/Direct)	Contract Type	Purpose / Responsibility
8	Token (USDPROP)	0x6F6c5Ab2865E028beDFEbabD86E046D73EAC5826	UUPS Proxy	Represents the real estate fund token. Controls minting, burning, and transfers under ERC-3643 rules.
9	NAVOracle	0x356B60F2600D0454871F9492a4B90cE750b0F9a9	UUPS Proxy	Oracle reporting the Net Asset Value (NAV). Restricts single-update variations to $\pm 10\%$ and cumulative daily changes to $\pm 20\%$.
10	SafeMultisig (Governance)	0xe1f7615962BcFEE1C9992A2c501702130835f92c	Standard (Safe)	Multisig vault (2-of-3 threshold) that holds system administration privileges and asset custody (Treasury).
11	InvestmentManager	0xe901ef0395850A217b19F2cc0819527ae4D2949f	UUPS Proxy	Manages USDC collection and USDPROP minting via EIP-712 cryptographic consent.
12	DividendDistributor	0x26FaD8d74c2Eb3461c94e5CeAC8e8AEb48e6D1D1	UUPS Proxy	Distributes USDC dividends to token holders based on historical balance snapshots.

3. Contract Interactions (Wiring Topology)

The USDPROP smart contracts operate interdependently to ensure that no transfer or minting bypasses KYC checks or financial boundaries:

Identity Module (ERC-3643 Core)

- The **IdentityRegistry** acts as the primary gatekeeper for the token.
- It queries **IdentityRegistryStorage** to verify if an address is bound to an ONCHAINID.
- It verifies in **TrustedIssuersRegistry** that signatures on the investor's claims originate from the authorized **ClaimIssuer**.
- It validates in **ClaimTopicsRegistry** that the investor holds the mandatory claims (e.g., Topic 1 for general KYC).

Compliance Module

- The **Token** delegates transfer validation directly to **ModularCompliance**.
- The **ModularCompliance** executes bound modules, primarily the **USDPROPComplianceModule**.
- The **USDPROPComplianceModule** verifies:
 1. Whether the investor is blocked by sanctions lists (OFAC).
 2. Whether country-level investment allocations (Country Caps) are exceeded.
 3. Whether regulatory holding lockups apply (365 days for Reg D / 40 days for Reg S).

Business Flow (Investments and Dividends)

- The **InvestmentManager** receives USDC from the investor, validates the EIP-712 signature, queries the active price in **NAVOracle**, and directs the **Token** to mint the corresponding amount of USDPROP.
- All received USDC is automatically transferred to the **SafeMultisig** (Treasury) address.
- The **DividendDistributor** receives USDC from the Safe and allows investors to claim dividends proportionally. It queries investor balance checkpoints directly from the **Token**.

4. Hardening and Control Handover (Governance)

Immediately following deployment, the setup script executes a **Hardening** phase to revoke temporary permissions from the deployer wallet (**deployer** EOA) and secure the system under the control of **SafeMultisig**:

1. **Agent Revocation:** The **deployer** EOA is removed as an agent from **IdentityRegistry** and **InvestmentManager**, and as a depositor from **DividendDistributor**.
2. **Ownership Transfer:** Ownership of all master contracts (**InvestmentManager**, **NAVOracle**, **DividendDistributor**, **USDPROPComplianceModule**, and **Token**) is permanently transferred to the **SafeMultisig** address.
3. **Role Segregation:** The administrative agent wallet (**adminAgent** EOA) retains operational permissions (such as pausing transfers or manually registering identities) but has no direct access to treasury funds or logic upgrades.