

Intuition Q4 2025

⌚ Pillar	Deployment
⌚ Priority	High
⌚ Status	In progress
≡ Source	Sales
⌚ Created by	👤 Nicolaus Sherrill
👤 Customer	<u>Intuition</u>
⌚ Linear Project	https://linear.app/decent-labs/project/intuition-custom-network-deployment-6097449e140b/overview
👤 PM	© Che Bajandas
⌚ Size	Medium

Overview

Intuition has their own network and, for legal reasons, wishes to support decentralized governance on their network ahead of their token launch. Towards this goal, they've deployed Safe contracts to their network, but would like to use Decent contracts for their governance operations.

Decent has not deployed to a new network in some time and is not currently poised to do so with ease. Although it may be relatively simple to deploy Decent contracts to a new network, it's not as simple to deploy the contracts of our dependencies and maintain support on the network indefinitely.

The timeline for this partnership is also tight, with a target date stated upfront as November 5th. As a consequence of all of the above, the team will focus on a narrow deployment to meet Intuition's minimum governance requirements.

Reference

Intuition Network Resources:

- [Testnet hub](#)
- [Mainnet hub](#)
- [Docs](#)

Objectives

- Deploy Decent contracts to the Intuition network
- Enable project creation on the Intuition network
- Block usage of features built on dependencies for which contracts will not be deployed to the Intuition network
- Get Intuition set up with a project they can begin to use for tokenized governance, using their token on the Intuition network as the voting token

Out of Scope

- Deployments required to support any of Decent's features that depend on third party contracts, including but not limited to: Hats, Sablier, 0xSplits
- Any crossover between Intuition's token on their network and a project deployed to a currently supported network like Base

Dependencies and Features Breakdown

Contracts we need to support / deploy:

- All independent Decent contracts: Fractal, Azorius, Linear ERC20/ERC721 Voting, Token Sales, Freeze Guards
- All required Safe contracts: Core wallet creation, multisig voting and execution, multi-send for batch transactions, asset management, Safe creation for subDAOs

- Core dependencies: Zodiac and OpenZeppelin

Note that Intuition already has Safe's contracts deployed to their network.

Also important: Decent's dependence on Etherscan. We'll need to work out hooking up Intuition's own explorer in such places where Etherscan is normally used.

Contracts we *do not* need to support / deploy:

- Hats Protocol and ERC6551 Registry (roles)
- Sablier V2 (payment streams)
- Disperse (airdrop)
- LightAccountValidator / EntryPoint v0.7 (account abstraction)
- 0xSplits (asset distribution/splits)
- Moralis (token balance and value)

Dependency ↔ feature breakdown:

Dependency	Features
Hats Protocol	Role based proposal permission
ERC6551 Registry	Role management Role payment streams
Sablier V2	Payment streams
Disperse	Airdrop
LightAccountValidator	Gasless voting
0xSplits	Revenue share AKA distribution
Moralis	Dollar value representation of asset holdings on app homepage, treasury, etc.
Subgraph	Project name lookup, Snapshot ENS lookup, project hierarchy

Features

Contracts Deployment

All contracts required for basic governance will need to be deployed on Intuition's network.

Intuition wishes to put forth proposals relating to contract changes and asset management. They would like for their token holders to have "veto power" on those proposals. In essence, they require basic tokenized (ERC-20) governance.

MVP for the Intuition deployment will be supporting the creation of a project (DAO) using the basic ERC-20 voting strategy, creation of proposals on the project, voting from token holders (allowing for rejection of a proposal through "no" votes), and execution of passed proposals.

Breakdown:

- Legacy contracts on Intuition network
 - Don't need new governance contracts (basically don't need anything that is unaudited)
- All, audited, new contracts on Intuition network
 - Token Sale, namely
- Zodiac module on Intuition network

API Updates

Once contracts are deployed, there will be some updates need to our APIs to support the new network.

1. Index the Intuition network
2. Hook up Intuition Safe API addresses

Potentially use new API integration to replace subgraph references, TBD based on frontend discovery.

Add Intuition to the Network List

There are at least two options for this:

1. Add Intuition to the network list permanently, as an option for anyone to use
2. Add Intuition to the network list as an option *only behind a feature flag*, allowing anyone with knowledge of the flag to create a project on the network

Option two would allow for either sharing the flag with the Intuition team, allowing them to do their own setup, or for Decent to do the setup for Intuition.

Decision:

Use option 2 from above for initial launch and play it by ear in terms of enabling permanently.

Block Unsupported Features for Intuition Projects



We'll be taking a "no design" approach here. Go for whatever we can accomplish without new UX layouts from the design team. The simplest, dev implemented UX will suffice.

Given that projects deployed to the Intuition network won't be able to use the full feature set, based on the minimal contract deployment that will be done, we'll need to block off access to those features that aren't supported and address any interface issues that might arise from the lack of certain dependencies.

The following full features / pages will need to be blocked off in the UX in some way:

- Roles
- Templates > Airdrop template
- Templates > Stream template
- Settings > General > Gasless Voting
- Settings > Distribution

Reconcile Dependence on Etherscan

Various parts of the app may be using Etherscan to collect information. This won't make sense for Intuition's network and instead such instances should use Intuition's own [explorer](#).

Most importantly, when creating a "from scratch" proposal, the target address lookup uses Etherscan to pull ABI options for the target contract.

Reconcile Dependence on Moralis

Moralis is currently used to fetch token balances and estimate token values.

We'll need to find another source of at least token balances to display in the app. Token value estimations are less important.

Potentially, we can use Intuition's explorer for this as well.

Replace Subgraph Usage

Subgraph is only used minimally in the app, but it is a dependency and does not exist on Intuition's network.

Ideally we use the Decent API to cover the functionality that is leveraging Subgraph right now.

decentSubgraph Metadata Fields:

- `daoName` - The human-readable name of the DAO
- `parentAddress` - Address of the parent DAO (for hierarchy support)
- `childAddresses` - Array of child DAO addresses (sub-DAOs)
- `daoSnapshotENS` - ENS name for Snapshot governance integration
- `proposalTemplatesHash` - IPFS hash containing proposal templates
- `hierarchy` - Recursive DAO hierarchy structure

Snapshot Subgraph Metadata Fields:

- Proposal data (title, body, choices)
- Voting records and results
- User voting power

Key Integration Points:

- **DAO Creation:** Subgraph data is indexed when DAOs are created
- **Dashboard Loading:** Fetches DAO metadata on page load
- **Hierarchy Navigation:** Uses parent/child relationships for navigation
- **Snapshot Integration:** ENS from subgraph enables off-chain voting
- **Treasury Operations:** DAO names and relationships for context

Out of Scope:

- **Proposal Templates:** IPFS hash from subgraph loads custom templates
- **Role Management:** Streams and role data for compensation