



© Che Bajandas 2025

Intuition Handoff Notes

⌚ Author	⌚ Che Bajandas
⌚ Category	Notes
👥 Team	💿 Product
⌘ Tag	
⌚ Last edited by	⌚ Che Bajandas
⌚ Last updated time	@November 6, 2025 8:25 AM
⌘ Status	In Review

What's being delivered

Project creation (including Multisig, ERC-20, and ERC-721 governance types) and basic governance functions. Basic governance includes:

- Proposal creation
- Proposal signing/voting
- Proposal execution
- Basic treasury operations (transfers)
- Project hierarchy/organizational structure management

What's been left out

- Role management (including role based proposal permission)
- Advanced payment features (airdrop and streams)
- Gasless (sponsored) voting

- Revenue splits
- Safe supported dApp “explorer” for proposal creation

Caveats

Lack of role based permissions

One of Decent's third party dependencies is the Hats protocol, which is the basis for the entire set of roles features. This includes the option to set proposal creation permission based on wallet address.

Due to time constraints, and the complexity of supporting third party dependencies on the Intuition network, these features are not currently supported.

In the meantime, the only way to control proposal permission on an ERC-20 token voting project is to use the token holding threshold permission. This defaults to allowing any token holder to create a proposal, but can be set to any amount.

Block delayed transactions

Many Decent contracts call a value defined as `block.number`. For Intuition, this value isn't actually the block number on the L3 network, instead it is the value on the parent L2 network (Base). Sequencers wait for enough transactions to accumulate on the L3 network before creating a block on the parent L2, or settlement layer.

This can mean a significant delay between confirming a transaction that was triggered by an in app action and the result of that transaction taking effect in the app, depending on activity on the Intuition network.

For example, in the Decent app, if you carry through with the transaction to create a proposal, the transaction completes quickly, but in the app you will not see your proposal appear promptly. Instead, based on the activity on the network, it could take several many minutes before it shows up.

Block-based time intervals

Decent's legacy contracts use block-based time intervals. There are updated contracts that use timestamp-based time intervals instead, but these have not

been audited, tested, or deployed in production. Block-based intervals was at one time the standard for Ethereum contracts, but has steadily been phased away in favor of timestamp-based intervals.

On Base mainnet, a block is mined roughly every 2 seconds. The Decent interface and governance system has been designed around this timing. The countdown for the voting period shows in app as a countdown of seconds, but this is only an estimation based on expected block mining intervals.

In other words, the timings displayed in the app on the Intuition network can potentially be unpredictable and fairly inaccurate at worst.

Network activity workaround

In testing, our engineers discovered that it's relatively easy to push through transactions and get new blocks mined in order to fast track transaction settlement.

Consistently, the team was able to follow up an app action based transaction with an *empty* transaction, and upon app refresh, the transaction is confirmed with another block having been mined. This can be done easily with your wallet by choosing the transfer option, selecting a currency, entering your own wallet address as a receiver, and entering zero for the amount.

Note that this really should only be necessary for governors, when setting up a new proposal or executing a passed proposal. Voters may not see their vote recorded immediately if network activity is low, but if network activity is high with other voters participating, the delay will be less apparent.

Intuition project (DAO) details

Objectives

Intuition wishes for a system of governance that provides token holders with a form of "veto power". Intuition will solely be responsible for putting up proposals that the community then has the opportunity to veto.

This isn't possible to achieve with a basic project structure. Instead, veto power comes in the form of "freeze" votes on a parent project targeted at a sub project.

The following section outlines how this would be configured.

Basic project configuration

Two parts of the project configuration need to be determined in order to fully satisfy Intuitions governance requirements.

For starters, the **parent project** needs to be configured. This will be a token voting project, using the veTRUST token as the voting token:

Field	Description	Value
Project Name	Name of your project/DAO	Suggestion: Intuition
Token Contract Address	Address of intended (ERC-20) voting token	0x635bBD1367B66E7B16a21D6E5A63C812fFC00617
Quorum	Percentage of total votes required for a proposal to pass	Suggestion: 3 or 4%
Proposal Permission Threshold	Minimum number of tokens a wallet must hold to create proposals (perhaps most importantly, veto proposals)	By default this is set to allow any token holder to create proposals. Think about what threshold you think should be reached to allow someone to propose a veto.
Voting Period	How long a proposal actively accepts votes	This doesn't actually influence veto votes. That's defined in the next section. Suggestion: 1 week
Timelock Period	How long a passed proposal is pending before it can be executed	Suggestion: 0; doesn't seem to be something that's needed for this use case
Execution Period	How long a proposal remains	This also doesn't apply to veto votes, just ordinary parent project proposals.

Field	Description	Value
	in the executable status before being marked as expired	Suggestion: 1 week

Next, the **child project** (AKA node) will need to be configured. This will be a multisig project with designated Intuition members set as signers:

Field	Description	Value
Project Name	Name of your child project/DAO	This likely should communicate something about this being Intuitions "governing body".
Signers	Addresses of designated signers	Whoever the responsible parties will be for creating and signing proposals!
Threshold	How many signatures are needed to pass a proposal	Depends on how hard or easy you want it to be for a proposal to pass. If you want this to be a one and done type of operation, make the threshold 1, then the creator of the proposal is enough to pass it.
Timelock Period	How long a proposal is locked down after it passes before it can be executed. This is the period of time where token holders have the opportunity to veto.	Think about how much time you'd like to give the community as well as how quickly you want your operations to go.
Execution Period	How long a proposal remains executable before it expires.	Keep in mind how long the timelock period and freeze proposal period are. If you'd like for a freeze vote to kill off the vetoed proposal, the freeze period should be longer than the timelock period and execution period combined.
Freeze Votes Threshold	How many voting tokens out of the total supply are	You may wish to match this value to whatever would be quorum on the

Field	Description	Value
	required to vote to veto before the project is frozen.	parent project.
Freeze Proposal Period	How long a freeze vote is active on the parent project.	This value should be a fair bit less than the timelock period to ensure that it has the chance to pass while the proposal is still timelocked.
Freeze Period	How long the project is frozen in the case of a successful veto vote.	If you'd like for the freeze to be long enough to kill off the vetoed proposal, make this value greater than the timelock and execution periods combined.
Enable Clawback	Whether or not the parent can clawback funds from the child project treasury.	TURN THIS OFF! This is not something that you want your token holders to be able to do.

Governance operations

In order to support a veto system, the project structure requires two aspects:

1. A parent project using token voting governance
2. A child project with multisig governance

The parent project's only purpose during normal governance operations will be to give the token holders the option to freeze the child project where actual proposals are drafted.

The child project will be set up with signers from Intuition and can pass proposals by meeting a simple signature threshold.

The way that veto works in this system is as follows:

1. A proposal is created on the child project
2. The proposal reaches the signature threshold to pass
3. The option to "timelock" the proposal is selected
4. The proposal enters the timelock period

5. If the community disagrees with the proposal, they can put up a vote on the parent project to freeze the child project
 - a. This can be done by anyone on the parent project with proposal power (by default, holding any voting tokens)
 - b. The option to freeze the project can be reached from the settings button on the child project
6. If the option to freeze the child project is selected, a proposal is created on the parent project
7. If the proposal on the parent project passes, the child project is frozen for the freeze duration
8. While the child project is frozen, proposals will fail to execute
9. If the freeze period is set long enough, the proposal may expire before it can be executed
10. As soon as the freeze period ends, proposals on the child project can be executed once again



One thing to keep in mind is that the parent project will be entirely controlled by token holders. If Intuition wishes to propose changes to it, they need to do so with large holdings of veTRUST. It also means that the community can propose changes to the parent, including something as innocuous as the name of the project, but extending to adding sub projects or changing governance settings. The parent cannot, however, make changes to the child project arbitrarily. They will not have any direct control over child project governance operations or funds in the treasury.

Future potential improvements

If desired, through varying degrees of effort, Intuition may opt to engage Decent to deliver further functionality to improve their governance operations:

- To improve the proposal permission, a role based permission structure could be supported

- Decent currently achieves this through a third party integration of the Hats protocol contracts
 - Supporting these features requires either building Decent contracts that cover these use cases or deploying Hats contracts to the Intuition network
 - Maintaining third party contract functionality on the Intuition network may require a higher degree of ongoing support
 - The feasibility and scope of this work is unclear at this time
- To improve time interval representation and mitigate potential related inconsistencies on the Intuition network, timestamp-based time intervals could be used
 - Decent's legacy contracts use block-based time intervals
 - Decent has updated contracts that have been ice boxed, are unaudited, and untested, that switch to using timestamp based time intervals
 - The scope and effort involved in upgrading to newer governance contracts is relatively well known and expected to be large since it impacts much of the existing app functionality and requires a migration strategy