# Feasibility Study - Data Availability with Validation on Cardano

Sky Protocol Team

2024-09-30

# Overview

> Sky Protocol is committed to constructing a tailored Data Availability Solution aligned with the requirements of Cardano. Our approach involves publishing data in a format conducive for Cardano smart contracts to seamlessly validate and integrate into Layer 1 contracts. Rooted in Cardano and underpinned by its own Cardano-based token, our network reduces reliance on external networks and minimizes trust assumptions. Our network enables multiple Cardano-based DApps to leverage the same Data Availability network, thus pooling resources and capital for network validation.

This report addresses Project Catalyst milestone 1 of the Sky Protocol: Data Availability for Cardano Layer 2 Solutions proposal.

# Criterion 1.a: Maintenance of Unique Identifiable Contract State

> How to maintain a unique identifiable contract state from one eUTXO to the next such that other contracts can read that state. This service may or may not already be provided by some Cardano contract languages, and may or may not be easy to provide on top of those languages. Presumably, underneath it all, an NFT identifies its holder as having “the” state for the bridge contract, and all lock scripts involved track which eUTXO possesses the NFT and what data is associated with it.

## Results

- For the maintenance of unique identifiable contract state, we will use NFT-based oracles, similar to the example from the Plutus Pioneer Program.
- Each data topic will be associated with an NFT that stores the root hash as its datum.
- Off-chain code will be used to find the NFT UTXO.
- We will use CIP-31 reference inputs to allow contracts to read the root hash.

# Criterion 1.b: Multisig for Data Availability Committee

> How to maintain a unique identifiable contract state from one eUTXO to the next such that other contracts can read that state. This service may or may not already be provided by some Cardano contract languages, and may or may not be easy to provide on top of those languages. Presumably, underneath it all, an NFT identifies its holder as having “the” state for the bridge contract, and all lock scripts involved track which eUTXO possesses the NFT and what data is associated with it.

## Results

- Initially, we will use a simple M-of-N multisig based on Ed25519 signatures.
- A configurable number of data operators will be required to sign hashes.
- In the future we might use Schnorr signatures for reduced size.

# Criterion 1.c: Merkle Proof of Data Availability

> How to verify in some smart contract language a “merkle proof” that data was present in a merkle tree the root of which was signed.

## Results

- We will implement the usual Merkle membership proof approach in Plutus.
- The path segments from the root to the leaf node in question (and of required sibling nodes) are included in the proof, and the root segment is signed.