

Object (RE) Oriented Debris 1 is part of a series of digital sculptures (2024-ongoing) inspired by the relentless orbit of space debris around Earth. These generative 3D sculptures evolve in real time, continuously shifting and reshaping into abstract configurations. Each sculpture features 10 unique modes of transformation, each inspired by the dynamic movements of Earth, space, and quantum phenomena. This constant metamorphosis immerses viewers in a vibrant, ever-changing digital environment, offering a rich multi-sensory experience.

By challenging the boundaries of dimensions, **Object (RE) Oriented Debris** invites audiences to look beyond surface appearances and question the fluid nature of reality. Every transformation contributes to an endlessly unfolding, machine-generated super-place that explores perception and the mutable essence of sculpture.

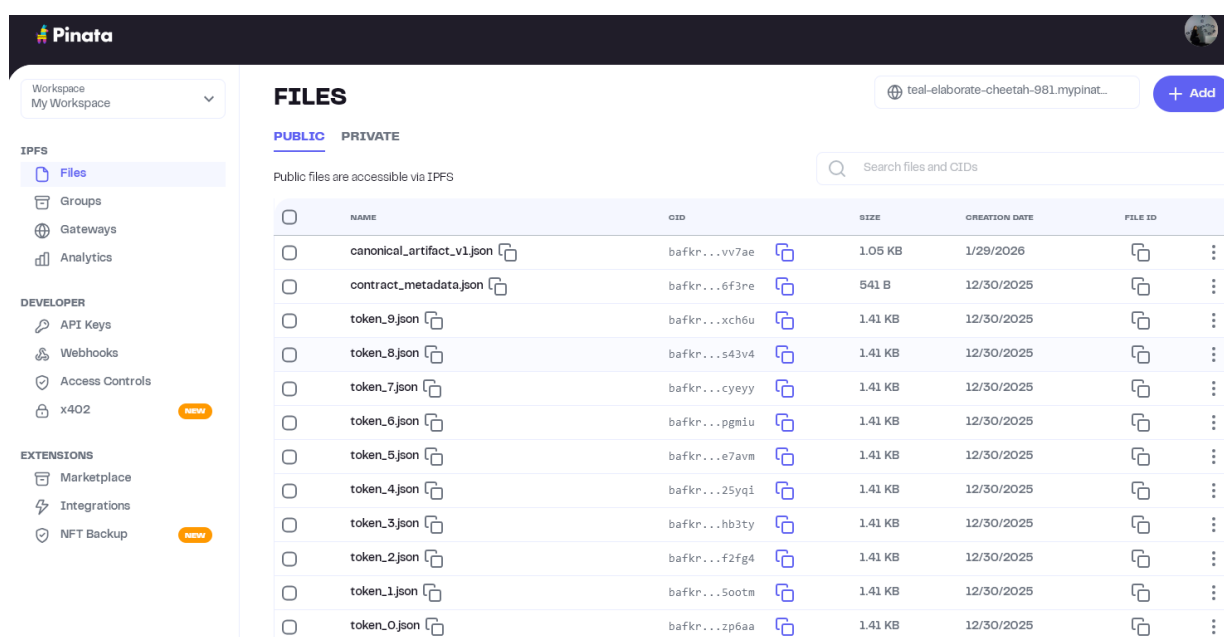
Interaction is integral: every time the **R key** is pressed (reload the browser page on mobiles), the sculpture shifts into a new behavior, triggering one of its **ten** distinctive **transformation** modes in an ongoing process of evolution. In an exhibition context, this interaction is further embodied through a customized interface in which the R key becomes a **physical, clickable sculptural element**, allowing visitors to trigger transformations through a tangible gesture that bridges digital action and material presence.

The concept is to edition each sculpture ten times, with each edition revealing a unique behavior. So, the same work exists in multiple, differing experiential forms.

Object (RE) Oriented Debris is accompanied by its own **SmartPy** certificate, a self-archival framework I developed to protect the work's preservation and guide any future restoration. Each of the sculpture's ten transformations and behaviors is documented in a publicly accessible JSON file.

Try the 1st one! <https://www.chiarapassa.it/ObjectOrientedDebris.html>

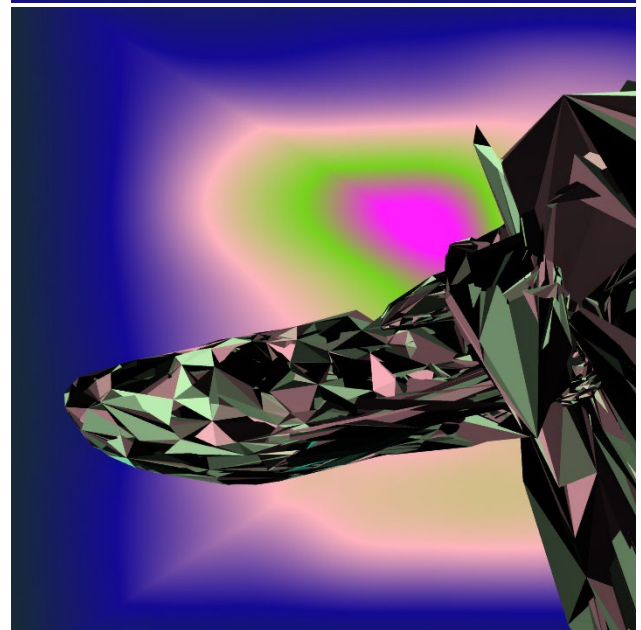
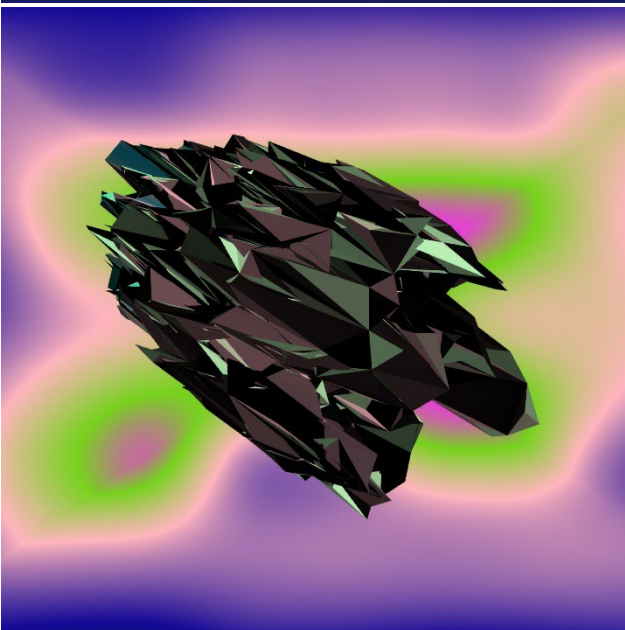
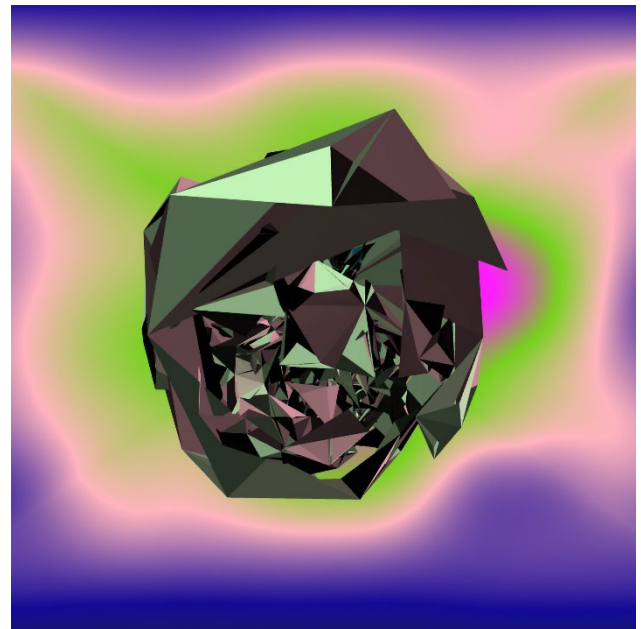
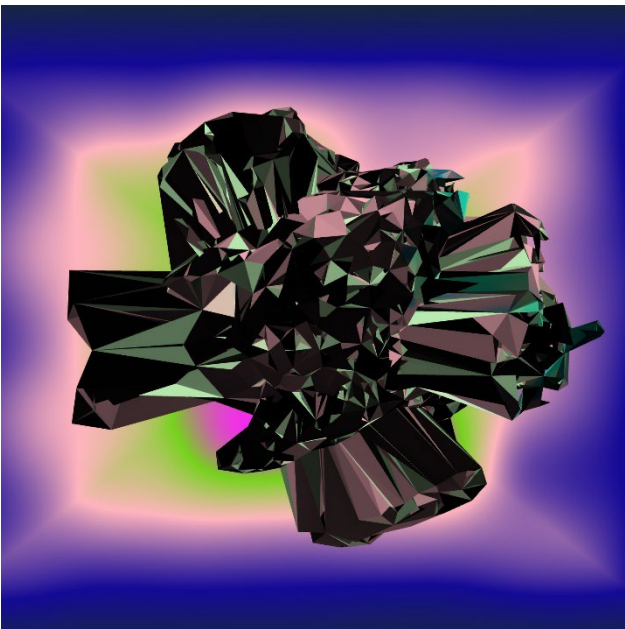
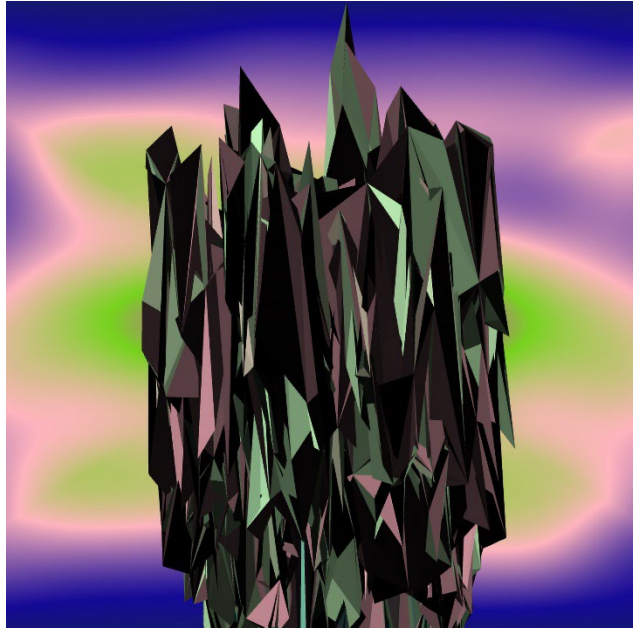
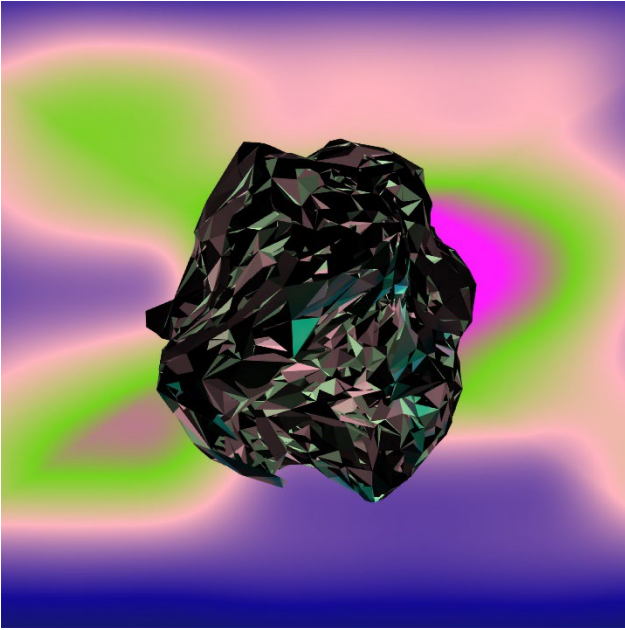
Video sneak-peek: https://www.youtube.com/shorts/nrU4b1iR_uM

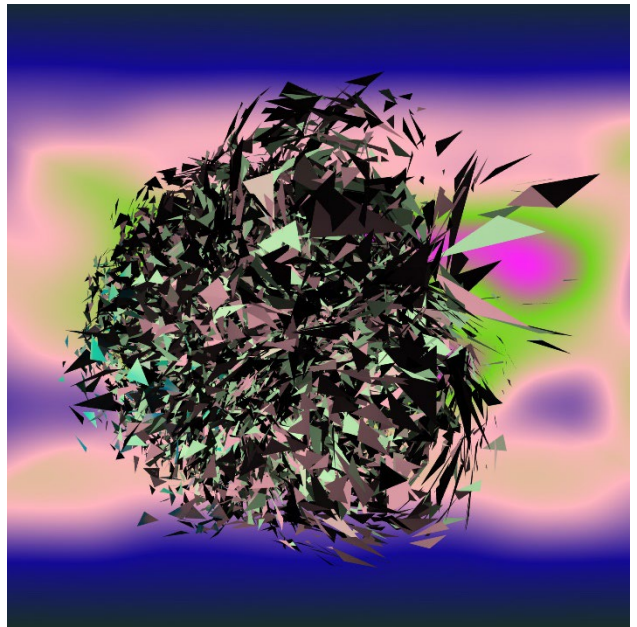
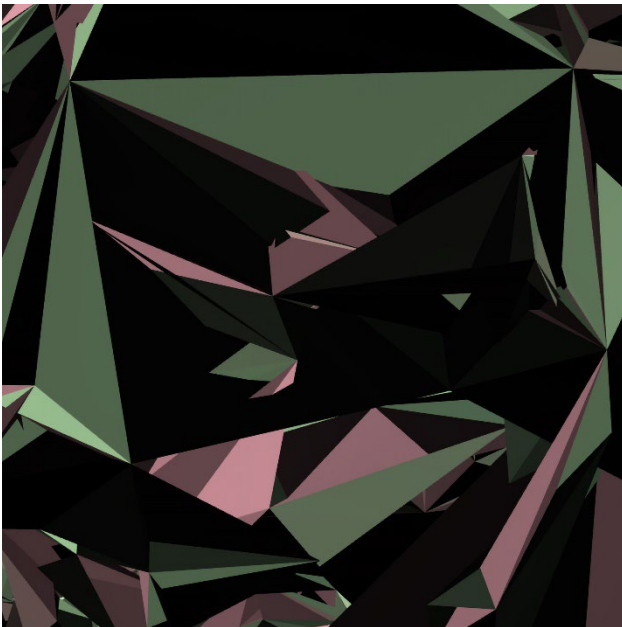
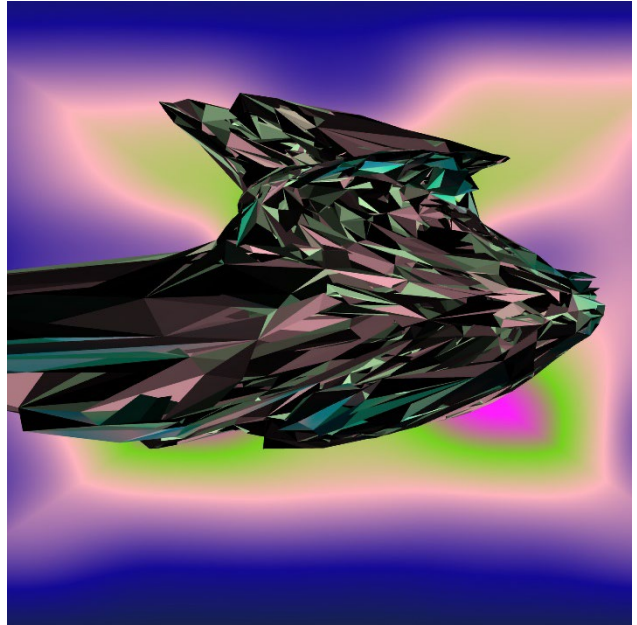
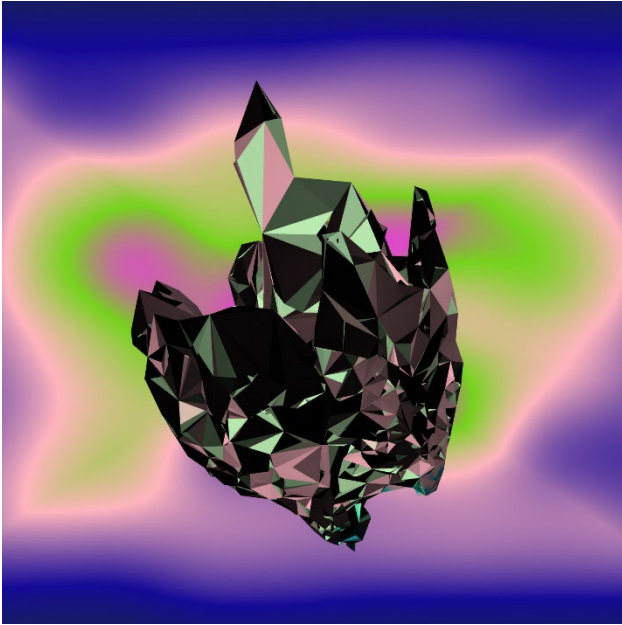


The screenshot shows the Pinata IPFS public storage interface. The main content area displays a list of files under the 'FILES' section. The files are listed in a table with columns for NAME, CID, SIZE, CREATION DATE, and FILE ID. The files are sorted by creation date, with the most recent at the top. The files are:

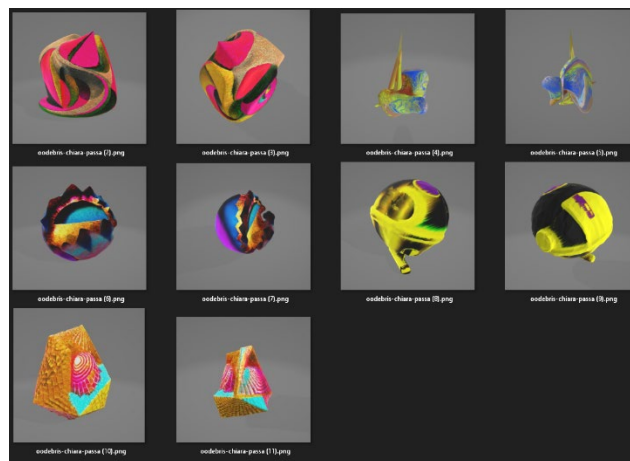
NAME	CID	SIZE	CREATION DATE	FILE ID
canonical_artifact_v1.json	bafkr...vv7ae	1.05 KB	12/29/2026	
contract_metadata.json	bafkr...6f3re	541 B	12/30/2025	
token_9.json	bafkr...xch6u	1.41 KB	12/30/2025	
token_8.json	bafkr...s43v4	1.41 KB	12/30/2025	
token_7.json	bafkr...cyeyy	1.41 KB	12/30/2025	
token_6.json	bafkr...pgm1u	1.41 KB	12/30/2025	
token_5.json	bafkr...e7avm	1.41 KB	12/30/2025	
token_4.json	bafkr...25yq1	1.41 KB	12/30/2025	
token_3.json	bafkr...hb3ty	1.41 KB	12/30/2025	
token_2.json	bafkr...f2fg4	1.41 KB	12/30/2025	
token_1.json	bafkr...5ootm	1.41 KB	12/30/2025	
token_0.json	bafkr...zp6aa	1.41 KB	12/30/2025	

Screenshot from the Json files on Pinata IPFS public storage





Screenshots from the generative 1st sculpture



Screenshot from the folder contains some other Object (RE) Oriented Debris digital sculptures

The Tezos smart contract I created manages a generative NFT designed for long-term conservation.

1. Minting

- The artist (admin) mints a token.
- Each token gets:
 - a unique `token_id`
 - an on-chain seed and generative parameters
 - an initial owner

2. Generative identity (on-chain)

- The seed and personality values are stored on-chain.
- These define the *identity* of the artwork and are immutable.
- The generative identity parameters are permanently written on-chain at mint time and cannot be altered, not even by the artist.

3. Artifact versions (off-chain, anchored on-chain)

- The artist can add multiple artifact versions to a token.
- Each version is an IPFS CID pointing to a JSON file (shaders, textures, engine version, notes).
- This allows technical migration (WebGL → WebGPU, etc.) without changing the artwork's identity.

4. Artist intent & conservation rules

- The contract stores the artist's intent (reinterpretation allowed, interactivity required, allowed or forbidden migrations).
- These rules guide how the work may be preserved or reinterpreted over time.

5. Restoration & conservation log

- Owners, authorized conservators, or the artist can log restoration actions via IPFS CIDs.
- This creates a transparent, on-chain conservation history.

6. Ownership & transfer

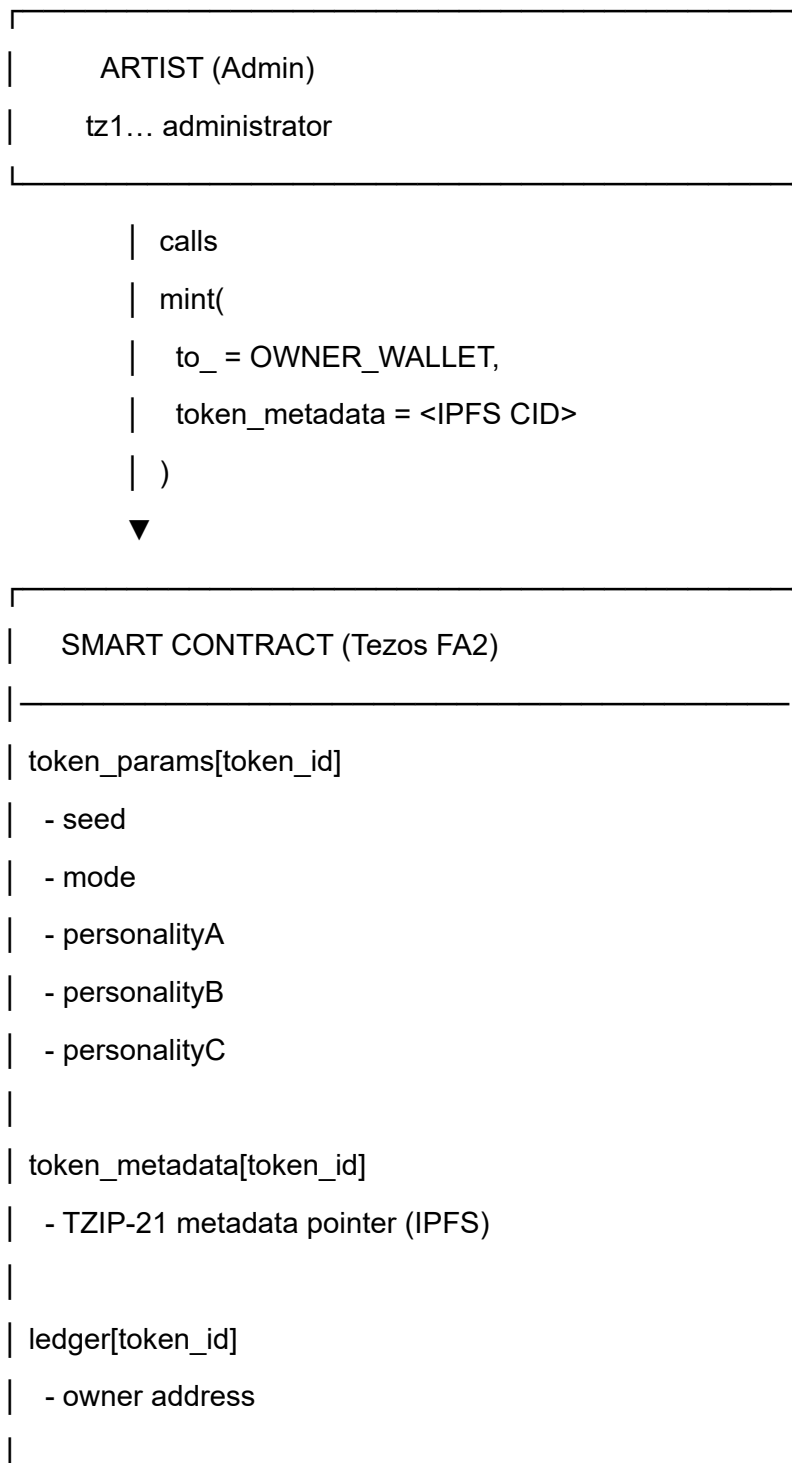
- The owner can transfer the token.
- Ownership is tracked on-chain independently from the artwork's technical form.
- The viewer can enforce these rules at the UI layer, for instance disabling prohibited interactions or migration types.

In short

The smart contract preserves the artistic identity on-chain, while the technical implementation remains off-chain and upgradeable.

This separation enables long-term conservation: the work can evolve with future technologies without compromising authorship, intent, or historical continuity.

SCHEME



| restoration_log[token_id]
| - list of IPFS CIDs (restorations)
|
| artifact_versions[token_id]
| - list of {version, ipfs_uri}
|
| GLOBAL ARTIST INTENT (storage)
| acceptable_reinterpretation : bool
| requires_interactivity : bool
| allowed_migrations[] : list
| forbidden_actions[] : list
| intent_cid : string (IPFS)

|
| add_artifact_version()
▼

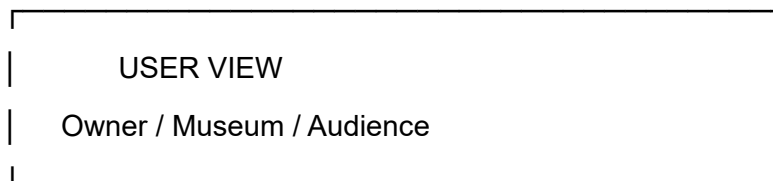
| IPFS STORAGE
| Canonical artifact JSON
| referenced by ipfs_uri
| may contain:
| - shader overrides
| - textures / assets
| - rendering parameters
| - conservation notes

|
▼

| JS VIEWER (WebGL / WebGPU Front-End)
|

- | A) Reads token_id from URL
- | B) Fetches on-chain token_params[token_id]

- | (seed, mode, personalityA/B/C)
 - | C) Fetches artist-intent fields from storage
 - | (reinterpretation, interactivity, migration rules)
 - | D) Fetches latest artifact_versions[token_id][-1]
 - | (canonical artifact CID)
 - | E) Downloads canonical artifact JSON from IPFS
 - | F) Applies technical overrides if present
 - | (shaders, textures, rendering rules)
 - | G) Enforces artist intent at UI level
 - Minted view → locked to on-chain params
 - Preview view → generative exploration allowed
 - | H) Renders the artwork deterministically
-



Chiara Passa, media artist exploring the digital dimensions of architecture and sculpture through augmented and virtual reality since 1997. I hold an M.F.A. from the Fine Arts Academy of Rome and a master's in Audio-Visual Media from the Faculty of Modern Literature. My artistic research is rooted in the mid-1990s revival of immersive art, where I explore software as a creative medium, from developing augmented reality applications to creating immersive, interactive virtual reality installations. My practice is deeply connected to a lifelong fascination with space as a mutable entity: a dimension that can be reshaped and reimagined through the language of informatics.

<https://www.chiarapassa.it/resume-chiara-passa.html> & <https://www.chiarapassa.it/cv-english.html>