

Art Factory

Media distribution and monetization network for
Artists, Curators and Customers

William Gleim

@billgleim

Art Factory: A ConsenSys Formation

Abstract

Solving the walled-garden problem of current content distribution channels, a decentralized mechanism for publishing, distribution and sale of digital content is presented. The ArtFactory network has been built, allowing content creators to publish, distribute and sell their work independently and permissionlessly. ArtFactory produces incentives [via a token economy] for the active participation of Curators (content distribution and discovery), Creators (creation and release of original works of art), and Consumers (of the created art via curators). ArtFactory presents a consumer-friendly platform where artist self-sovereignty is protected by payment and file distribution that is handled by code and not third parties.

Keywords: Ethereum, Blockchain, Tokens, Content Publication

1. Introduction

2 This paper describes Art Factory. Art Factory is an anti-fragile framework
3 for distribution and monetization of original creative art.

Email address: `bill.gleim@consensys.net` (William Gleim)

4 **2. The ArtFactory**

5 *2.1. Mission Statement*

6 The Mission Statement of the Art Factory is to persist an anti-fragile art
7 store with incentives for creation, discovery and consumption.

8 *2.2. Components*

9 The implementation of Art Factory builds on the foundation provided
10 by decentralized system components including Ethereum, Infura, MetaMask
11 and IPFS.

12 *2.3. Requirements*

13 The requirements for the Art Factory are:

- 14 • persist a durable product framework for creation, distribution and mon-
15 etization of original creative works
- 16 • incentivize the participation of Creators, Curators and Consumers

17 *2.4. Design*

18 The design of ArtFactory is presented in its entirety in **Appendix A**.

19 **3. ArtFactory Token Launch**

20 *3.1. Requirements*

21 The requirements for the ArtFactory token launch are:

- 22 • provide opportunities for individuals to purchase tokens representing
23 publish, curation and purchase rights within ArtFactory
- 24 • provide utility for community members organized around the shared
25 goal of fostering a rewarding landscape for the creation and publication
26 of original art

27 *3.2. Design*

28 The ArtFactory token launch generates and issues tokens representing
29 rights to publish, curate and purchase content within ArtFactory.

30 A minority percentage of tokens will be reserved for the creators of Art-
31 Factory. After the initial token sale period, token holders may elect to publish
32 and purchase content in ArtFactory.

33 Further details of **token launch**, **proceeds allocation** and **token al-**
34 **location** will be forthcoming.

35 3.3. Implementation

36 The Art Factory token launch implementation builds upon best practices
37 of token sales from the Ethereum ecosystem.

38 The Art Factory and Art Factory Token Launch contracts have been
39 implemented and deployed and are live on the Ethereum testnet. Public
40 deployment to the mainnet and kickoff of the Art Factory token sale are
41 pending resolution of smart contract finalization and security review.

42 Preliminary tokens can be purchased with test Ether for demonstration
43 purposes. For every 0.001 tETH sent to the Art Factory token launch con-
44 tract address on the testnet, 1.00 testnet ArtFactoryToken will be given in
45 return. For clarity, the testnet tokens have no value. The testnet tokens
46 have no association with the tokens that will be issued at the time of the
47 subsequent mainnet Art Factory token sale.

48 Appendix A. Art Factory

49 Here we provide a brief overview of Art Factory requirements, actors,
50 protocol, implementation and usage.

51 Appendix A.1. Background

52 ArtFactory is currently implemented as a single-blockchain platform oper-
53 ating on the Ethereum blockchain, employing the Interplanetary File System
54 (IPFS) for file storage. Publishers and purchasers of content hold Ethereum
55 and ArtFactoryToken (AFT) tokens in Ethereum wallets.

56 Appendix A.2. Requirements

57 The explicit requirements for Art Factory consist of

- 58 • publish content
- 59 • purchase content
- 60 • consume content

61 Additionally, Art Factory includes the following **implicit** requirements

- 62 • discover content
- 63 • promote content

- 64 • allow consumers to explicitly pay content publishers
- 65 • allow consumers to implicitly pay content promoters and curators
- 66 • allow blacklisting of content

67 *Appendix A.3. Actors*

68 The actors in the system are:

- 69 • Content Creator
- 70 • Content Publisher
- 71 • Content Consumer
- 72 • Content Curator

73 For the discussion we operate on the assumption that the Content Creator
74 is the same actor as the Content Publisher. Violations of this assumption lead
75 to questions of copyright infringement. The mechanism to address copyright
76 infringement on the platform is a blacklist for disputed content. Blacklists
77 are generally enforced by the Curator actor.

78 *Appendix A.4. Protocol*

79 The ArtFactoryContent protocol provides the ability for each user type
80 to interface to the ArtFactory platform to perform their respective roles.
81 Provided here is a simplified form of the protocol specification.

82 1. Structures

- 83 • ArtFactoryContent

84 2. Methods

- 85 • publish
- 86 • pay
- 87 • view
- 88 • withdraw

89 *Appendix A.5. Implementation*

90 The ArtFactory protocol is exposed through smart contract interfaces
91 implemented in Solidity for the Ethereum Virtual Machine (EVM).

92 *Appendix A.5.1. Publication*

```
function ArtFactoryContent(string _name, uint _price, string _handle);
```

93 *Appendix A.5.2. Payment*

```
function pay() payable returns (string);
```

94 *Appendix A.5.3. Consume*

```
function getHandle() constant returns(string);
```

95 *Appendix A.6. Usage*

96 ArtFactory includes a lightweight HTML/JS front-end that facilitates
97 Publication, Discovery and Consumption of content. The lightweight HTML/JS
98 front-end is provided as a reference implementation for Curators.

99 Example usage of the smart contract interface to the ArtFactory protocol
100 is provided in the remainder of the section.

101 *Appendix A.6.1. Publication*

```
ArtFactoryContentContract.new( "content_0xAA", 10000000, "QmPpeiynkfVpTajH3CMb8682  
newContent_0xAA = ArtFactoryContentContract.at("0x7a6550b6fd5edd33d2cca48f4f7fd454
```

102 *Appendix A.6.2. Payment*

```
newContent_0xAA.pay({from: web3.eth.accounts[0], value: 10000000, gas: 1000000}, f
```

103 *Appendix A.6.3. Consume*

```
newContent_0xAA.getHandle({}, function(error, result) {if (!error) contentHandle =
```

104 *Appendix A.6.4. Get Name*

```
newContent_0xAA.name({}, function(error, result) {if (!error) contentName = result
```

105 *Appendix A.6.5. Get Price*

```
newContent_0xAA.price({}, function(error, result) {if (!error) contentPrice = resu
```

106 *Appendix A.6.6. Retrieve Purchased Status*

```
newContent_0xAA.purchased({}, function(error, result) {if (!error) contentPurchase
```

107 *Appendix A.6.7. View Balance*

```
newContent_0xAA.viewBalance({}, function(error, result) {if (!error) availableBala
```