DArt: Blockchain applied to Cultural Heritage

Claudio Di Ciccio[1], Giordano Dionisi[1] & Michele Spina[1]

[1] *La Sapienza University of Rome, Department of Computer Science, Faculty of Information Engineering, Computer Science and Statistics, Italy*

eMail / di.ciccio@di.uniroma1.it, dionisi.1834919@studenti.uniroma1.it & spina.1711821@studenti.uniroma1.it

Abstract / **DArt** is a *decentralized application* based on **Algorand/Ethereum** for *Culture Heritages.*
DArt proposes to apply **Blockchain Technology** to change the managment of any type of *Cultural Asset* in the real world.
Any useful information and update about *Artwork* can be written in **DArt,** and each user can consult it; constituting an element of *Study, Analysis and Monitoring* for the *Art World.*
Any Artwork is represented by a **Token** and using **Smart Contracs' Technology** is possible to limit token writtability to only *Verified Sources*⋆.
QUESTA PARTE SI POTRA' SCRIVERE MEGLIO PIU' AVANTI NEL PROGETTO

Keywords / Blockchain, Distributed Ledger, Distributed Systems, Ethereum, Cultural Heritage

1. Background

This section would introduce you some useful knowledge and notion to understand the paper and how DArt works. These knowledge are the starting point of DArt project.

1.1. Needed knowledges

1.1.1. Distributed Systems

A distributed system is a computing environment in which a collection of autonomous computer systems that are physically separated, communicate and coordinate actions in order to appear as a single coherent system to the end-user.[4] Therefore a distributed system must be resistant to delays, crashes, failures and in some cases even to the byzantine faults, if one or more participants can act against the interest of the system.[8]

1.1.2. CAP Theorem

According to the Cap Theorem, any distributed system can provide only two of the following three guarantees:

- Consistency: Every read the most recent or receive an error;
- Availability: Every request receives a non-error response, without the guarantee that it contains the most recent write;
- Partition tolerance: The system continues to operate despite an arbitrary number of messages being dropped or delayed by the network between nodes.

Usually in distributed systems we adopt availability and partition tolerance, discarding consistency.The same is valid for the blockchain.[6]

1.1.3. Hashing

An hash functions is a function used to map a data of arbitrary size to fixed-size values. The input values is called key and the outputs are called hash values or hashes. Two fundamental properties of an effective hashing are the speed of calculation and a minimized probability of collisions, that is, of finding two different keys that generate the same output. Thanks to these properties, hashing can be used to generate pseudo-random numbers. From this chapter on, when we refer to a hash function, we will refer to a hash function generating pseudo-random numbers.

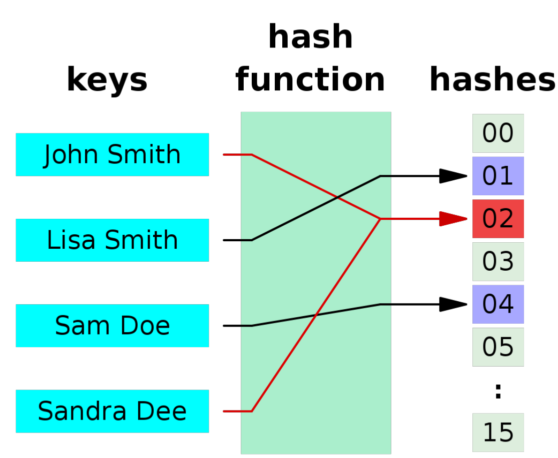

Figure 1: Hash Function

In other words, given one key k_1 and one hash function h, it is possible to compute $v_{k_1} = h(k_1)$ with very low computational cost, but given h and v_{k_1}, it is not possible to reconstruct h^{-1}, therefore calculating k_1 has a very high computational cost, because you need to hash every possible key k_i, until k_1 or a colliding key is found. The hash value v_{k_1} is a pseudo-random value, because it's generate by a deterministic function applied to a fixed key, but its value is hardly predictable.

1.2. Blockchain

A blockchain is an open and distributed ledger. It's a chain of blocks composing a connected and acyclic graph with only one leaf. Each block can contains many transaction that represents the information in the ledger. The transaction and the blockchain structure are immutable.

1.2.1. Bitcoin

Previously we have told what is a blockchain, that it is basically a mathematical structure.

On this simple concept many protocols have been developed, like as Bitcoin, to manage money, firstly, but then to manage transactions that need to be done and need to be untouched. The bitcoin is the most diffused application of blockchain technology. In fact it is the first most famous blockchain technologies.

Briefly we have that blocks are linked together using hash functions and we have that we pay using transactions. When a user has a transaction to publish, then he has to give it to a miner. It creates the block accumulating all these transactions and it has to hash all the entire block. The resulting hash needs to have with a certain amount of zeros at the end. If this doesn't happen, then the block isn't correct, otherwise it is exactly correct and it can be published. So we use the hash function only to avoid that all miners publish a lot of blocks together, but to have a temporization, practically.

Specifically Bitcoin is a protocol and to see how it works we see the following thing:

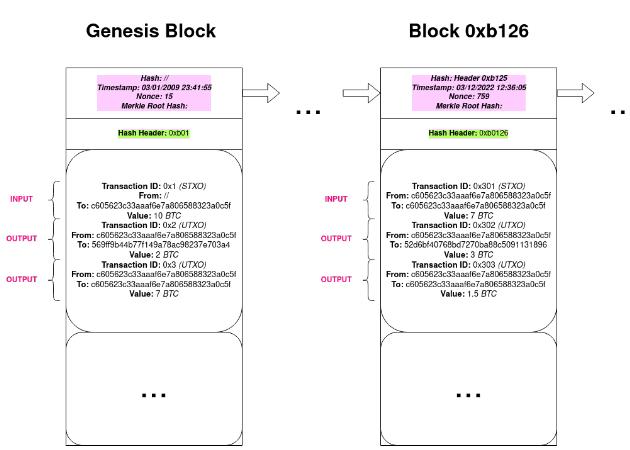

Figure 2: Costruttive Example

We can see that we have the following structure, where we have transactions that we use to spend money or not, namely each block has:

- An Header Part, structured as following:
  - Hash of Previous Block to maintain the chain;
  - Timestamp of the block: The timestamp for miner to understand when it's mined that specific block. Not really useful, because we are in a distributed environment and a global clock doesn't exist, but it's needed for the other nodes to understand if a block is really valid or not;
  - Nonce: It is the only parameter that the Miner can change to obtain the desired hash. It is only an integer;
  - Merkle Root Hash: It's an hash of the list of transactions of the current block;
- Hash Header: We need to hash the header of the block;
- Body of Block:
  - For each transaction we have:
    - A transaction ID: Identifier of transaction globally;
    - From Field: The sender of each transaction;
    - To Field: The receiver of each transaction;
    - Value: The value in BTC to transfer.

    These fields are for the one input and the two outputs of each transaction:

    (i) The input is the transaction that we need to spend (STXO), so it is like as the banknote;

    (ii) The first output is the transaction to transfer to the receiver, so it is like as the banknote to transfer and to give to a specific user;

    (iii) The second output is the "rest", so the amount of money that they remain from the input and the first output, in fact it's a transaction that needs to be spent for the user (we split transactions, practically);

    (iv) If we have:

    second output input - firts output

    Then we have that this remain part is for the miner, so it is the "commission" to execute our transaction.

All this is the basic principle of Bitcoin and the basic concept of the Proof of Work that it's based on proof by means of a computational effort.

1.2.2. Ethereum 1.0 (Proof of Work)

In this case we haven't anymore that the transactions can be spent like as Bitcoin, because we have a total different concept: Account Model (or Balance Model) and not Transaction Model as Bitcoin.

In practise we don't exchange transactions, so we haven't anymore transactions to be spent for each entity, but we have a real amount for each entity to be spent, so we exchange values.

In every node we have installed Ethereum Virtual Machine (EVM) to run the smart contracts (that we will see after). Each transaction is executed by each specific node and using the EVM we have a standardization of execution having always same environment for all executions, so we have a middle-layer to avoid compatibility's problems between different CPUs, Systems and so on. The EVM is so useful to do operations in the Blockchain for Ethereum.

Each account in Ethereum is represented by means of a table (so many rows) and in each of them we have stored the balance (expressed in wei) of the specific Ethereum account.

In Ethereum we associate a cost for each basic instruction that we can perform: Specifically we have that in Ethereum we deploy Smart Contracts, so pieces of code to execute things; this means that these Smart Contracts are codes, so instructions and these instructions can be translated, with the EVM, in a EVM bytecode → For each one of these instructions we can associate a cost. This cost is expressed in *gas* and this last one do never change in time, respect to Ether (that it fluctuates according to trading). The gas price is needed because every smart-contract needs to be executed by any node in the Ethereum context and we want pay the computation made to all these nodes. Let's suppose to have the following situation:

Bob's transaction with:

(i) Gas Price 200 Wei
(ii) Gas Limit 12.000 gas units

In this way we are telling that each gas unit corresponds to 200 Wei, so if it is too low like as amount (if it is not convenient) then the miner can decide to ignore and to not consider it, exactly⋆. Instead the gas limit is the maximum amount of gas that our code can consume for us, so to avoid that we do a code that does an infinite loop or something similar (so to avoid that all our balance goes down), we can specify a threshold over that the execution of smart contract will end, so we know the maximum amount of Ether that we need to spend, in worst case.

We can also have the following:

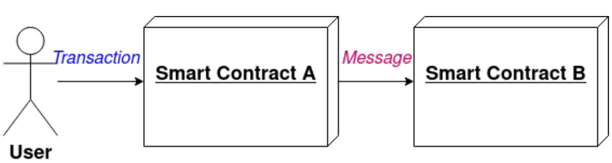

Figure 3: An Example for Gas Price

In this way we can easily understand that the User calls the Smart contract A and then this last one calls the Smart Contract B: Even the costs to call the Smart Contract B are paid by the User at same way.

The transaction will be recorded in the ledger, instead the Message will not be recorded, because we can compute the result after the execution of it, because, theoretically, any node knows its code and so everything regarding of it.

If we have errors or not, then we have that the miner earns the gas regarding of its made execution and the remaining part (so the gas not consumed by the execution) comes back to user.

⋆*For this reason first of all any transaction will be executed firstly in mining nodes, because they will include our transaction*

We have states in Ethereum and we pass from one state to another one each time that a transaction will be executed. If we have n transactions in a block, then we have that each block has n changes between transactions and for each transaction we pass from one state to another one, according to how many instructions will execute. In case of errors we roll back and the miner doesn't get the reward (only the specific fees).

The difficulty of the Proof-of-Work in Ethereum 1.0 increases every 100.000 blocks and this means that we will arrive in a moment that the difficulty to forge a block is too high that it's needed a lot of time to do this forging and so the validation of the blocks for miners becomes really too high, in terms of time, and this will bring all the system to die. Also it will not more convenient to take the reward forging a block for the computational effort needed, so.

In Ethereum we have three main types of rewards for miners:

(i) Reward: It is given by the system for the work done to forge a new block;
(ii) Fees: They are given by the users, so each transaction has it. It is needed to get a priority for the transactions by the users;
(iii) Ommers Reward: This is a particular case:

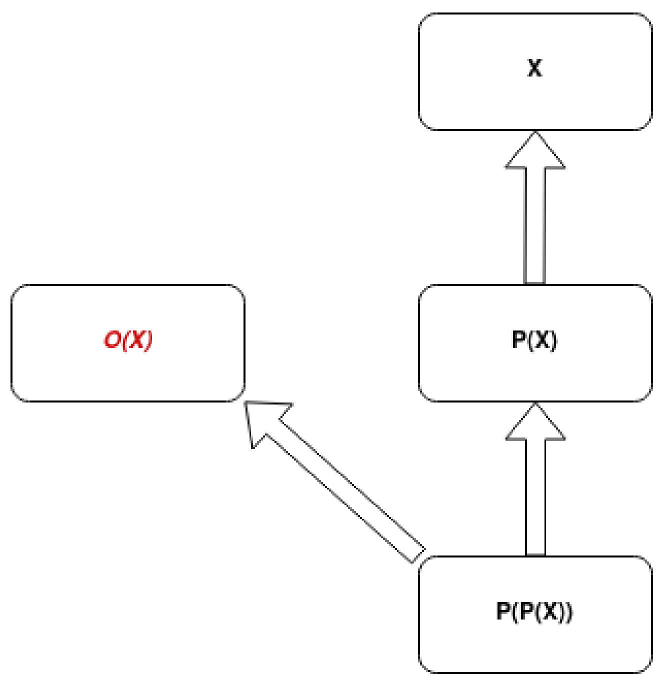

Figure 4: Ommers Reward

In fact we have that there is a fork in a certain moment (this can happen even in Bitcoin, when two different miners win the game more or less simultaneously). The fork that will die (because the maority will continue the other one) is rewarded. We have an higher reward if we are an ommer of first generations, otherwise the reward will be reduced. Only one time we can take the ommer reward for a specific block.

Two different types of nodes we can have:

(i) Full / Archival: These nodes store everything of the blockchain, all transactions, states and so on. With these we can see the actual state of the BlockChain, avoiding recomputations;
(ii) Simple Nodes: They do only some controls, checks and similars.

Two types of chain we have:

(i) Chain Data: This type of BlockChain contains all blocks with the specific transactions that all these blocks constitute entirely the chain. The Chain Data is needed to see the correctness of hashes (to avoid changes, so);
(ii) State Data: It's a smaller version of the BlockChain and it contains only all the results of each state transition of each transaction.

1.2.3. Ethereum 2.0 (Proof of Stake)

1.3. Web 3.0

The blockchain and the Internet have an high correlation and blockchain technology is really interesting for Internet and we have that:

(i) At begin Internet had clients and servers and we also had requests from clients to servers in a client-server model → The computational effort for nodes was really a lot low compared to computational effort for Databases (so we had a cross of things);
(ii) In web 2.0 we had an high computational effort (so high computation) done by local machines (so an high power for nodes);
(iii) In web 3.0, so nowadays, we have a merge between blockchain topic and internet topic and both them are merged together → In the blockchain we have stored only important things and out-of-chain we have pictures and so on. So blockchains aren't totally needed, but they are really a lot useful.

2. Context

This chapter describes the context of application of DArt, the developed dapp and the subject of this paper, i.e. the world of cultural heritage management and its digitization, in order to better understand the needs and potentials.

2.1. Evolution of management of Cultural Heritage

The idea to take care of a state's artistic heritage (that it's a completely contemporary goal) it is often very widespread, but this does not correspond to the historical truth. Humanity has manifested interest on management of cultural heritage since ancient age, and since the XV century the pope has designated a specific cardinal for the only purpose of managing the registers of works of art belonging to the Papal State to verify their movements and status, with the title of cardinal chamberlain.[13] Our approach to cultural assets has totally changed in the last centuries and still today academics and art critics discuss about it. An Example: only in 2019 many academics have raised criticisms about the over-stress of Vitruvian Man, involves by transportation and continuous exhibitions in different museums and galleries that compromise irreversibly its ink.

The idea of how an artwork should be made available and managed is constantly evolving and presents differences from culture to culture even today in which we live in a highly globalized world, despite being much closer to an international standard. The collaboration from different countries and the cultural Exchange have permitted management of cultural heritage to involve, fueling the debate, and in many case saving very important cultural Heritage from irreversible damage or even destruction.

Also the restoration techniques are continuously evolving, using new chemical products and observing their effects in the time. keeping track of which compounds have been applied to a work, how long and in which conditions it has been preserved, such as humidity, temperature and lighting, are fundamental elements for understanding the behavior of these products and progressing in development of better restoration techniques.

2.2. Third parts controllers

The system that regulates and controls the management of cultural heritage Is developed on three levels. UNESCO - United Nations Educational, Scientific and Cultural Organization, established in Paris on 4 November 1946 - acts as an international guarantor and organizer of conferences through which the member States stipulate pacts and conventions. The World Heritage Convention, dating back to 1972, is in fact the first international instrument that defines the notions of protection and preservation of cultural heritage.[16]

Although UNESCO's domain is very extensive, its politicized nature means that today even some large countries do not partecipate, like the US and Israel withdrew in 2019.[3]

At the second level we see, in Europe's case, the European Union which protects and preserves the cultural heritage of the member states through initiatives and incentives. Indeed, the Council of the European Union adopts measures and recommendations to encourage the action of the Member States, which are responsible for their own cultural policy matters. It is in fact the single State, at the third and last level, which must enact legislation, in compliance with international treaties.

We than see external bodies which, due to their international prestige, act as controllers and promoters of the protection and enhancement of cultural heritage. Among the most important we see ICCROM, an intergovernmental organization that works at the service of its member states to promote the conservation of all forms of cultural heritage, in every region of the world and ICOM, the International Council of Museums, or the main international non-governmental organization representing museums and their professionals. The organization assists the museum community in preserving, conserving and sharing present and future cultural heritage, tangible and intangible.[2]

2.3. Digitization in the World of Art

In the last decade, digitization in the world of culture has become an important subject of study and research, developing fields such as digital museology and digital literacy to make cultural heritage more accessible to anyone. In addition to giving us the ability to remotely access entire libraries or the ability to digitally view an art gallery, technology is also revolutionizing the way cultural heritage is archived by digitizing what were once huge physical paper ledger around the world, subjected to errors, wear and tear and difficulty in consulting. In particular, in recent years there have been questions about how the blockchain can impact on the management of the registers of cultural heritage, so much so that already today it is possible to find several papers on this topic online.

3. Goal

The goal of the project is to build a public ledger using the blockchain to take trace about the management of each cultural heritage. Contrary to many modern projects, DArt does not propose itself as a digitization of artistic material, but as a tool for managing it in the real world. The goal of DArt is not to impose a digital point of view on art, but to put modern technologies at the service of humanistic and artistic culture for the management of cultural heritage according to what is modern ethic. The intent of DArt is to constitute a support element for actors in the art world to migrate towards the international modern definition of museum:

> A museum is a not-for-profit, permanent institution in the service of society that researches, collects, conserves, interprets and exhibits tangible and intangible heritage. Open to the public, accessible and inclusive, museums foster diversity and sustainability. They operate and communicate ethically, professionally and with the participation of communities, offering varied experiences for education, enjoyment, reflection and knowledge sharing - *ICOM, August 2022*[11][7]

3.1. Actors

The DArt's goal is to be a verified source for any person to analyze and study the management of art in world, looking at museum's practice, The information in the ledger are written only by verified source like museum, art gallery, private collectors and restoration groups. Each person can read these information and look at the source. On the ledger will be possible to analyze information about any artwork in the chain, and also look at any action of the actors. Each user will also have the opportunity to be a patron of any museum through donations bound by smart contracts.

3.2. Applications and uses

The applications of DArt in the management of cultural heritage can be many, in this section some of them are proposed in order to understand the possible impact of DArt on cultural heritage world.

- **Protection**: It consists of any activity aimed at preventing, protecting, maintaining and restoring the assets that make up our cultural heritage with the aim of being able to be publicly enjoyed by the community.[12] DArt will allow to keep track on the blockchain of all the protection activities carried out on an artwork and their key information.
  - **Preventing**: Any action to remove a possibility of risk.
  - **Protecting**: Limitation of risk situations connected to the cultural property in its context
  - **Maintaining**: Coherent, coordinated and planned study, prevention and maintenance activities. Contrary to restoration, maintenance is a preventive action, and tends not to affect the object. Part of this process is monitoring stress level on a work of art.
  - **Restoring**: Dart allows you to keep track of all the restoration works carried out over time, allowing you to verify the good work of the managers and in the case of future restorations, to know which chemicals have already been applied on the work, in order to avoid unwanted reactions.
- **Exhibitions**: Using DArt it will be possible to keep track of where a work has been exhibited, but at the same time know the history of museums and art galleries, thus giving a tool for a critical analysis of them and to measure the overexposure and underexposure of the artists, for example knowing which galleries host the most exhibitions that give more space to female or non-Western artists.
- **Forgery**: Having a trace of the status over time of a work of art can help make its falsification more and more complicated.
- **Donations and Patrons:** Users have the possibility to make restricted donations to the verified actors, tying their use to the protection of a specific artwork. Donor users will be able to be certified through an NFT and obtain the advantages as patrons, established by the institution receiving the donation.
- **Tracciamento passaggi proprietà**: ciao...

4. Design

In the **Design Part** we have the need to understand *How Design* and *How See* the *Environment* and the *System* for an *Abstract Point of View.*

4.1. Use Cases

The **Use Cases** are one of the most important part of our project, because they explicit which are the *Possible Actions* and *Functions* for our *Users.*

4.1.1. RequestId

Always we need to add *ArtWorks* in the *Blockchain.* This can happen according two different cases:

- **New Artwork** created by an *Artist,* so we need to **Store** it;
- An **Old Artwork** put in *blockchain* only now, but it is an old one. This can happen because before there isn't any other insertion in the *Blockchain* (no **Digitalization** of it).

So we need to check that the Artwork isn't already in the *Blockchain* and then we insert this *Artwork* in it. In this way we compute an **Hash** of the name of the *Artwork* to have a **Global Identifier** of it and it's also important to *Memorize* the correct *Position* of the *Artwork* in that specific moment (*eg.: in Louvre in Room A6 third place*).

We can show you the image of **Diagrams** for this *Use Case,* namely the **Flowcharts** and the **Other Ones** (we will do this only for this *First Use Case,* the other ones you will find in the *Appendix*):

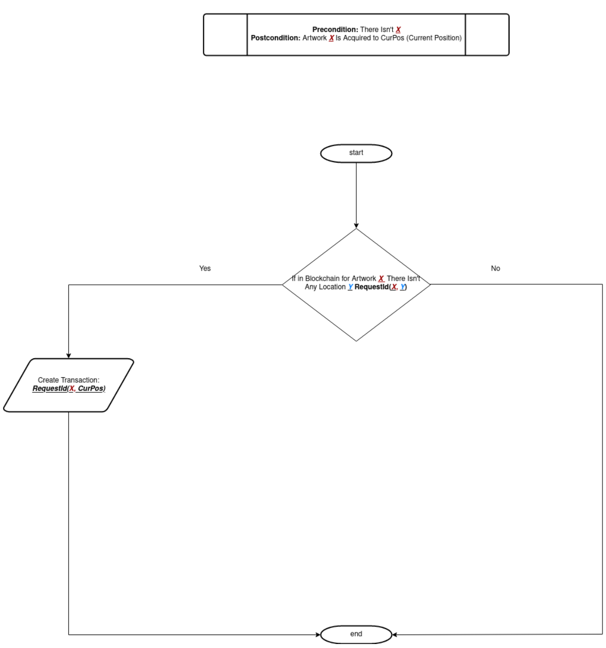

Figure 5: Request Id FlowChart

4.1.2. Restoration

DArt allows you to *Keep Track* of all the **Restoration's Works** carried out over time, allowing you to:

- *Verify* the *Good Work* of the *Managers*;
- In case of *Future Restorations,* to know which *Chemicals* have already been applied on the work, in order to *Avoid Unwanted Reactions.*

4.1.3. Stress

DArt can be used to Track the **Stress Factors** to which an *Artwork* has been subjected, thus being able to measure the *Level and Predict or Measure the Implications.*

4.1.4. Exhibitions

Using **DArt** it will be possible to *Keep Track* of:

- **Where** an *Artwork* has been *Exhibited;*
- *Know* the **History** of *Museums and Art Galleries,* thus giving a **Tool** for a *Critical Analysis* of them and to *Measure* the *Overexposure and Underexposure* of the *Artists***

  We can do two Possible Things:

- *Put* the *ArtWork From* the *Warehouse* of a specific *Place* $\delta^{\star\star\star}$ to the *Exhibition Place of* δ
- Some time we need to *Remove* the *Artwork From the Exhibition Place of* δ and this means that then we leave the *Artwork* in the same *Place* δ*,* but we put it in the *Warehouse, for e.g.* so *Hidden for Normal Visitors.*

4.1.5. MoveId

We need to **Move** the *Artwork* from one *Place* δ to another *Place* δ' (*Temporarily or Permanently*) and we *Identify* the *Artwork* with the usual *Hash Value* of if, obviously.

4.1.6. Forgery

Having a Trace of the *Status Over Time* of an *Artwork* can help make its **Falsification** more and more *Complicated.*

4.1.7. Property Passage

DA SISTEMARE

It's important to understand that we have two important things to fix: namely we have that an artwork can be in a specific place and it can be of a specific entity: so we have two different concepts, namely the location where an artwork is and the property assigned for that specific artwork. So for example we can have that:

- Monnalisa is of property of the Louvre Museum, instead it is exposed in the Prado's Museum.

Obviously we have an high problem, in fact a thing that we need to fix is that only the who has the property of the artwork can do everything, instead who has in its collection the artwork cannot do everything, because the artwork isn't really of its property. So we need to do some permissions according to who can do the actions and who cannot do them.

For the property's passage we have the follow things:

if the actor**** γ wants to give a certain artwork to another actor γ', then we need to check if γ has really

***e.g.: Knowing Which Galleries host the most Exhibitions that give more Space to Female or Non-Western Artists, to Understand Where it's better to go.*

****e.g.: Museum*

****e.g.: Museum

this artwork and we need to pass some rights, that we need to understand, for this specific purpose.

In this case we have a property's passage, so we need to understand for which reason an entity has this need to sell a property, so maybe we have that we can also monitor the money passage for the acquisition of the artwork. We can have that each entity has the need to be authenticated by an external service (like as the Spid for the normal users) and each entity has an amount of tokens to but artworks and so on. Obviously each entity has an amount of "NFTs" to understand which artworks are of a specified entity γ.

ADD: We can have also an idea more clever so that we are the central unit that certify the various artworks of all the γs.

5. Strategy

In the previous section what the issue are that DArt wants to resolve, so in this part of the report it's described the strategy choices to solve this needs, choosing the best technology for this use case.

5.1. Technology

From the analysis made we can deduce that one of the main needs is to manage a large amount of data, coming from actors scattered all over the world. Cultural assets travel from country to country and can be managed by both private and public entities. This raises the need to establish a close relationship of trust and collaboration between the actors of art world.

The development of a **decentralized application** on blockchain was therefore chosen to meet these needs.

5.1.1. Smartcontract

5.2. Methods

5.3. Example

6. Development

We have explained in the *Appendix* the reasons of **Why** we need a *blockchain,* **Which** *Type of blockchain* is more *Indicated* and **Which Specific Type** of *blockchain* is more Related to our *Purposes.*

6.1. Transactions For Use Cases

We have different types of possible transactions for any possible block, in fact we can have transaction to specify if an artwork was moved, if it is gone in restoration amd so on. So the most important things to see are:

- Restoration: It is a specific use-case and action that we can have for a specific artwork and for it we have that we insert from determined entities if or not a specific artwork is going to do in restoration **or** it's finished, in which date, what it is done, for which reasons, the amount of money needed and so on. So a new transaction of this type we need to have when a new restoration begins or when a restoration ends;
- Damage: We have that when an artwork has a damage (strong or mild) then we need to understand what it's damaged, in which forms, timestamp, what we need so to do, money needed and so on. With it we can suggest a successive restoration, but it is not mandatory. Obviously a restoration can be linked to a certain damage;
- Exhibitions: We have a transaction when an artwork is exposed in a new place (a new city, a new room, a new museum and so on)
- Forgery: We have an inserted transaction in the Blockchain to avoid that we can have fake artworks and to permit this we need some identifiers on the same artworks (for example the representative signs, dimensions of artwork and so on). So, in other words, we have some key informations or we can add the correspective hash value of the artwork hidden in the artwork, for example behind it. In this way only the entities that can can work on the artwork can know the correspective hash Value (or a key value) and these entities are trusted, so we assume that they don't tell this important identifiers to anybody. A more interesting thing that we can add is an adhesive that it has a value to understand the uniqueness of the artwork and this adhesive is putted behind the artwork or under or in a place that hasn't problems for the persons. This adhesive has an unique value not known and it is signed by the director of the museum (or similars) that has the original one (initially the original one has to be certified by professionists, obviously). Obviously this adhesive and the correspective sign can be hidden, it is preferrable, as much as possible;
- Grant Permissions: Obviously it's possible that a certain entity sells a specific artwork or that it wants to grant some permissions also to other specific entities because, for example, a specific artwork goes in exhibition to another museum and so we need to grant to this other entity the permissions to create transactions for the specific loaned artwork. To do this we can implement a functionality in the Smart Contract, called only from the specific original museum, so
- Generation of a New Artwork: When we have a new artwork to insert in the blockchain, we need to be sure that we have the original one (certified like as usual. We have to build and to call the Smart Contract for the Blockchain of the Artworks to insert precisely it.

We have also that we can compute and establish the stress of each artwork according to the restoration made, the damages and so on!

7. Software Architecture

References

[1] *UNESCO culture for development indicators: methodology manual.* UNESCO Publishing, 2014.

[2] *Museums, Ethics and Cultural Heritage.* Taylor & Francis, 2016.

[3] Thomas Adamson. U.s. and israel officially withdraw from unesco, Jan 2019.

[4] Krzysztof R. Apt. *Verification of sequential and concurrent programs / Krysztof R. Apt, Ernst-Rudiger Olderog*. Texts and monographs in computer science. Springer, New York [etc, c1991.

[5] Vitalik Buterin. Ethereum white paper: A next generation smart contract & decentralized application platform. 2013.

[6] Seth Gilbert and Nancy Lynch. Brewer's conjecture and the feasibility of consistent, available, partition-tolerant web services. *SIGACT News*, 33(2):51–59, 2002.

[7] ICOM Italia. Definizione di museo: Scelta la proposta finale che sarà votata a praga.

[8] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Trans. Program. Lang. Syst.*, 4(3):382–401, jul 1982.

[9] Lorenzo Mucchi, Matilde Milanesi, and Claudio Becagli. Blockchain technologies for museum management. the case of the loan of cultural objects. *Current Issues in Tourism*, 25(18):3042–3056, 2022.

[10] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system. May 2009.

[11] International Council of Museums. Museum definition.

[12] M.A. Sandulli. *Codice dei beni culturali e del paesaggio*. Le fonti del diritto italiano. Giuffrè, 2012.

[13] Salvatore Settis. Le radici romane della tutela del patrimonio culturale. *L'Osservatore Romano*, page 1–5, Nov 2009.

[14] Ziyuan Wang, Lin Yang, Qin Wang, Donghai Liu, Zhiyu Xu, and Shigang Liu. Artchain: Blockchain-enabled platform for art marketplace. In *2019 IEEE International Conference on Blockchain (Blockchain)*, pages 447–454, 2019.

[15] Amy Whitaker, Anne Bracegirdle, Susan de Menil, Michelle Ann Gitlitz, and Lena Saltos. Art, antiquities, and blockchain: new approaches to the restitution of cultural heritage. *International Journal of Cultural Policy*, 27(3):312–329, 2021.

[16] L. Zagato, S. Pinton, and M. Giampieretti. *Lezioni di diritto internazionale ed europeo del patrimonio culturale. Protezione e salvaguardia*. Libreria Editrice Cafoscarina, 2019.