This article has been accepted for publication in IEEE MultiMedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MMUL.2023.3303393

Encoding of Media Value Chain Processes Through Blockchains and MPEG-21 Smart Contracts for Media

Mirko Zichichi*†, Víctor Rodríguez-Doncel†
*IOTA Foundation, Germany
mirko.zichichi@iota.org
†Ontology Engineering Group, Universidad Politécnica de Madrid, Spain
vrodriguez@fi.upm.es

***Abstract*—Distributed Ledger Technologies can be used for rights management in the audiovisual production sector, where dominant business models still need to provide a sustainable way to support the claims of content creators and rights holders fully. This paper describes the combination of the current set of MPEG-21 multimedia framework standards with Distributed Ledger Technologies and smart contracts. Their gathering shapes the Smart Contracts for Media, a specification that can be used to encode the terms and conditions of a contract for media-related delivery and consumption. We provide the implementation of a system based on the Smart Contract for Media to enable the twofold process of reducing the complexity of contract terms compliance validation and making stakeholders more aware of the media value chain. We also provide the technical details for a Video-On-Demand Services use case in which the exploitation of media rights is modeled through the MPEG-21 framework and the Smart Contract for Media. Finally, we perform an evaluation of our smart contracts implementation by analyzing the gas cost. Results suggest the viability of our approach.**

***Index Terms*—MPEG, Distributed Ledger Technology, Smart Contract, Non Fungible Token**

I. Introduction

MPEG (Moving Picture Experts Group) has developed several well-known media encoding international standards for audio, video, and genomic information. One of its endeavors is the definition of the Multimedia Framework, known as MPEG-21, which is an open framework for delivering and consuming multimedia.

The newest part of the International Standard is the Smart Contract for Media (MPEG-21 SCM), which enables the use of Distributed Ledger Technologies (DLTs) and smart contracts to address a set of very well-defined challenges [1].

DLTs can easily fit into the MPEG-21 framework to exploit their advantages. These technologies can act as resonance boxes for instances created using the MPEG-21 framework and directly enforce what has been determined in terms of use of the media [2]. The features DLTs provide can reduce the opacity of complex systems processes: (i) transparency, for the append-only ledger is auditable by the whole DLT network; (ii) immutability, as data cannot be easily tampered with; (iii) traceability and non-repudiation, because each network participant cryptographically signs each transaction issued in the immutable ledger and (iv) decentralized execution of immutable instructions, i.e., smart contracts.

Smart contracts are software procedures that can be run to ensure the proper execution of new types of applications directly on a DLT. In some implementations, a smart contract can be considered a specific interpretation and translation of some contractual terms. However, the mere fact that a smart contract is stored on a DLT does not give rise to a legal agreement: legal contracts need to comply with a complex hierarchy of laws and regulations at the local, national and international levels, which may limit the ability of a smart contract to give rise to a legal agreement and the scope of their enactment. In combination with the MPEG-21 framework, smart contracts can be used to encode the terms and conditions of a contract for media-related asset trading. Indeed, rather than passing through centralized intermediaries, revenue from a stream or download of media content could be distributed automatically to rights holders through a smart contract.

The objective of this work is threefold:

- Our main objective is to demonstrate how to exploit the Smart Contract for Media to manage IP rights and administer media royalties in the specific context of the MPEG-21 framework. By bringing together the MPEG-21 multimedia framework and smart contracts we aim to provide the following advantages: (i) to foster transparency in the media value chain and reduce disputes over the royalties distribution; (ii) to provide automatic mechanisms for the execution of the agreements; (iii) to reduce the complexity of validating compliance with contract terms and conditions, including limiting revenue losses due to contract violation and illegal distribution of content; (iv) to enhance disintermediation and render those directly affected, e.g., content creators, more conscious about the media value chain.
- Second, we aim at providing a detailed description of the Smart Contract for Media, its relation with the MPEG-21 media value chain, and a possible implementation using a DLT, i.e., Ethereum. In our implementation description we will go through the components needed to build and execute a SCM, namely a MPEG-21 contract parser, the

This article has been accepted for publication in IEEE MultiMedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MMUL.2023.3303393

Ethereum DLT, key manager and the InterPlanetary File System (IPFS).

- Finally, we provide a use case based on Video-On-Demand rights management through the Smart Contract for Media and a performance evaluation of its smart contracts implementation. The use case demonstrate how smart contracts can be used to establish and enforce IP agreements such as licenses and enable the transmission of real-time payments to IP owners; IP rights information in protected media content, then, can be encoded using the MPEG-21 framework and directly and uniquely linked to a smart contract, i.e., a Smart Contract for Media. In other words, smart contracts can be used to allow music and video media royalties to be administered almost instantaneously and manage usage allowances and restrictions.

The remainder of this paper is organized as follows. Related work is presented in Section II. Section III describes the Smart Contract for Media, Section IV provides a possible implementation. Section V describes a use case related to video-on-demand services' media value chain and appraises the performance evaluation of its smart contract implementation. Section VII provides the concluding remarks.

II. Background and Related Work

A. Distributed Ledger Technologies and Smart Contracts

A DLT consists of a set of protocols and components that guarantee untampered data availability, where nodes in a network mutually agree on a shared state while tolerating failures and malicious behavior to some extent. The append-only ledger is extended through transactions that are disseminated throughout the network and that are independently verified by each node in order to ensure their consistency. A smart contract is a code deployed on a DLT or the source code from which such code was compiled, whose execution of its immutable instructions is distributed among the nodes of the DLT it is deployed to. Each node executing the instructions receives the same inputs and produces the same outputs, thanks to a shared protocol. These properties allow the issuer of a smart contract not to require the presence of a trusted human third-party validator to check the terms of an agreement.

The decentralized applications that are possible to build on top of DLTs thanks to smart contracts, exploit the verifiability of information stored on the distributed ledger and authentication based only on cryptographic primitives. The token representation is one of the most used. It is information recorded on a DLT representing some form of right: ownership of an asset, access to a service, receipt of payment, etc. The Non Fungible Token (NFT) is a utility token usually implemented to represent and transact with (tangible or intangible) assets on DLTs, where every single token is different from the rest of the tokens, i.e., non fungible. More specifically, NFTs combine both concepts of (i) access rights to an underlying economic value (property), and (ii) permission to access someone else's property or services or collective good [3].

B. Semantic Web and Rights Expression Languages

Semantic Web technologies bring structure to the meaningful contents of the Web by promoting common data formats and exchange protocols. The most spread paradigm to represent information is RDF (Resource Description Framework). In this framework, resources are identified with URIs and described with collections of triples. The precise meaning of each resource can be formally established with ontologies. An ontology is a formal representation of knowledge through a set of concepts and relations between these concepts within a specific domain. Through these ontologies, it is possible to convey the meaning of data, facilitating cross-domain applications and services. Rights Expression Languages are a central component of contemporary digital rights management systems. They are applied to express permissions, obligations, and prohibitions in a machine-readable form. The authors in [4] propose a classification to understand their functionalities and applications, giving an outlook on how these languages are used to explicate machine-readable rights for access control, trust management, and contracting.

C. MPEG-21 Contracts Representation

The MPEG-21 framework includes languages and ontologies that facilitate the conversion of media narrative contracts to digital ones and enable the creation of new contracts in machine-readable electronic formats.

Media Value Chain Ontology (MVCO) [5] is an ontology used to describe the main entities in the media value chain formally: (i) IP entities, which are the objects subject to copyright law protection such as works (e.g., an original song), manifestations (e.g., its music score), instances (e.g., the performance of the song), or products (e.g., a sellable item); (ii) relevant actions that can be performed on those entities (e.g., adapt an original work, perform a specific work), and (iii) types of users whose actions are rights, obligations, or something else provided by IP law (e.g., creator, producer).

Media Contract Ontology (MCO) builds on MVCO's generic deontic statements by providing the elements to shape the structure of media contracts, to express rights to exploit media content and to define specific obligations for payments and notifications [6].

In the MVCO and MCO cases, the use of RDF is involved. However, the MPEG-21 also includes a part involving the eXtensible Markup Language (XML), i.e., the Contract Expression Language (CEL) [7]. CEL provides an extensible model for representing generic agreements between parties and defines the most common acts and constraints in the media field and is used in digital media contracts.

D. Related Work

Many academic works have proposed using DLTs and smart contracts to manage content-related rights. For example, Lee et al. [8] [9] propose a private blockchain to manage digital artworks and AI models, with a GDPR-compliant, Hyperledger Fabric based implementation. Other solutions focus on the reduced consumption of resources [10], on works registration

This article has been accepted for publication in IEEE MultiMedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/MMUL.2023.3303393

TABLE I
LIST OF ACRONYMS

CEL	Contract Expression Language
DLT	Distributed Ledger Technology
IP	Intellectual Property
IPFS	InterPlanetary File System
MCO	Media Contract Ontology
MPEG	Moving Picture Experts Group
MVCO	Media Value Chain Ontology
NFT	Non Fungible Token
RDF	Resource Description Framework
SCM	Smart Contract for Media
VOD	Video-On-Demand

or on fighting counterfeiting. However, not many of them have placed the focus on interoperability, being neither blockchain-technology agnostic nor relying on international standards. The issue of inadequate compensation for "artistic effort" in traditional platforms has prompted numerous firms, such as Ujo Music and Vezt, to explore the use of DLTs in developing innovative ventures within the music industry [11]. However, existing implementations primarily focus on basic mechanisms for royalties management using so called utility tokens and offer limited solutions for media rights management.

In contrast to these existing works, this paper aims to emphasize the expressive potential provided by the Smart Contract for Media within the MPEG-21 framework, rather than simply building an ecosystem. A notable related work is the Open Digital Rights Language (ODRL) [12], a recommendation from the World Wide Web Consortium (W3C) that leverages Semantic Web technologies to facilitate the distribution, sharing, and exploitation of statement information related to content and services. However, this still lacks a direct integration into a DLT environment. To address these gaps, scholars have explored the integration of Semantic Web technologies with DLTs to enhance content management. For instance, Banerjee et al. [13] propose an architecture that utilizes DLT to provide a purpose-centric access-control model, using Semantic Web technologies to create new ontologies for policy declaration. The authors in [14] propose the PrivDRM (Privacy-Preserving DRM) system, which enables consumers to obtain DRM-protected content and its license while safeguarding their personal data. However, these two works lack comprehensive solutions for media rights management.

In summary, existing works discussed above have primarily focused on token-based royalties management, lacking solutions for media rights management and interoperability of policies and metadata. The methodology we propose aims to address these technical gaps by emphasizing the expressive potential of the SCM within the MPEG-21 framework.

III. THE SMART CONTRACT FOR MEDIA

The Smart Contract for Media specification is a passthrough component designed to be the interlingua that connects the MPEG-21 framework with different DLTs. Thus, it was designed to contain a set of tools interoperable with different types of DLTs and for use in different contexts inherent in the

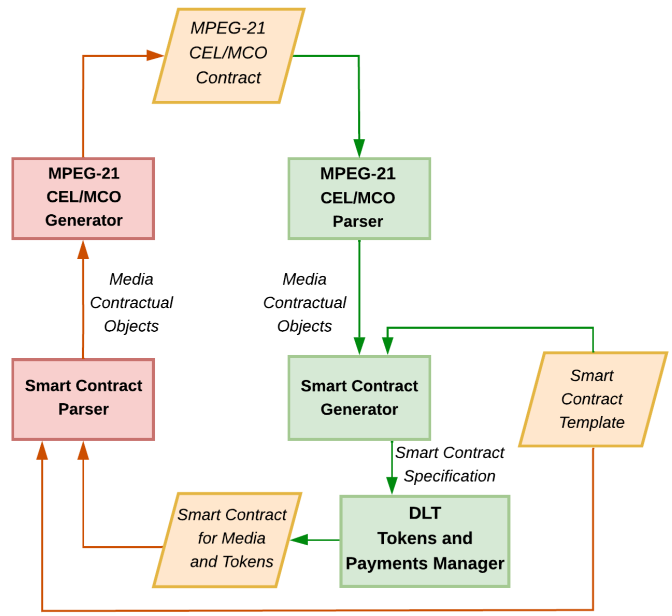

Fig. 1. Smart Contract for Media conversion from (green path) and to (red path) an MPEG-21 CEL/MCO Contract. Box shape represent a system component, while diamond shape an artifact or a result of a computation.

media value chain. Rights information is protected content, then can be encoded using the MPEG-21 framework and directly and uniquely linked to a smart contract, i.e., an SCM. In other words, smart contracts could allow content policies to be administered almost instantaneously and manage usage allowances and restrictions.

A. Conversion from/to MPEG-21 Media Contracts

The MPEG-21 machine-readable contracts expressed in CEL and MCO/MVCO (that, from now on, we will refer to them as MPEG-21 CEL/MCO) are based on the common structure that narrative media contracts share. An MPEG-21 CEL/MCO Contract includes a preamble extracted from a narrative contract version with: a unique identifier, date, version, title, *Parties* of the contract, and references to other contracts that are being amended or substituted by this. The MPEG-21 CEL/MCO Contract object also includes the body with contract clauses called *Deontic Expressions*. For instance, a *Party* has an obligation of payment after broadcasting some media as specified in one of the contract's permission.

Figure 1 shows a flowchart on how the conversion process starts from (green path) an MPEG-21 CEL/MCO Contract (top yellow box) to obtain a SCM (bottom yellow box) and viceversa (red path). The conversion process is as follows:

- Firstly, when someone wants to create a new SCM (green path) the starting point is a MPEG-21 CEL/MCO Contract. This is parsed by a MPEG-21 CEL/MCO Parser component to obtain a set of Media Contractual Objects. This kind of object is described in the next sub section.
- The Media Contractual Objects are given in input to the Smart Contract Generator component together with a

Smart Contract Template. This template is created previously to dictate the behavior of the SCM once deployed to the DLT. The Smart Contract Generator component finally outputs a Smart Contract Specification.

- Lastly, a DLT Tokens and Payments Manager component takes as input the Smart Contract Specification and deploys a new SCM on the DLT.
- The original MPEG-21 CEL/MCO Contract can be re-obtained directly from the SCM (red path) using the Smart Contract Parser component first and then the MPEG-21 CEL/MCO Generator component.

B. Media Contractual Objects

The SCM is a smart contract that includes or refers to metadata and contractual information connected to creative works, i.e., media, and encodes a contract's terms and conditions. These metadata are called Media Contractual Objects. The following describes what are the Media Contractual Objects and how are they used in the SCM.

1) Parties: The *Party* object represents a human or juridical person bound to the narrative contract. Since identities in DLTs are generally represented through addresses, a *Party* is represented and authenticated in the SCM through a DLT address that, thus, represents this *Party*.

2) IP Entity: The *IP Entity* object encapsulate one or more digital items of intellectual property in the MPEG-21 multimedia framework, such as an original work or a music performance. In the SCM, *IP Entity* objects are uniquely identified on-chain through the use of NFTs. Then, the entire set of information related to a specific *IP Entity* object is linkable to such NFT.

3) Deontic Expression: The *Deontic Expression* encompasses the properties of an agreed machine-readable contract clause regulating parties' actions and rights, such as permissions, obligations, and prohibitions. The uniqueness of such an agreement leads to following the same approach used for *IP Entity* objects, i.e., clauses are serialized according to the concept of NFT. The reasons for supporting this approach are: (i) it enables a unique way for storing clauses in DLTs, that is also beneficial in terms of interoperability, in terms of sharing these clauses with other DLT-based applications; (ii) it allows the transfer of value in the form of obligations, permissions and prohibitions, similarly to how cryptocurrency transfers are done.

C. Smart Contract and Distributed Ledger Technology

From a practical point of view, the SCM can be considered an interface that makes the MPEG-21 framework interoperable with several DLT implementations.

1) Instructions: One of the roles adopted by the SCM is to directly and passively enact what is "enactable" (i.e., enforceable) in a DLT, with reference to the clauses indicated in the media contract. The clauses indicated in the media contract are represented as *Deontic Expression* objects, and the implementation of instructions in the form of smart contract methods are derived from these. For instance, a specific *Payment Deontic Expression* object leads to the creation of a specific payment smart contract instruction.

2) NFTs and Immutable Storage: The second role of the SCM is to crystallize the data encoded using the Media Contractual Object. This is due to the native immutability feature that the DLTs' ledger generally provides.

Each Media Contractual Object is stored in the SCM according to what was discussed in the previous subsection. In particular, each *IP Entity* and *Deontic Expression* is stored in a unique NFT To be noted is also the fact that the MPEG-21 *Contract* object preamble might include the narrative contract text version, too, in the form of an object or at clause level, making thus explicit the legal isomorphism.

We stress that NFTs are already used for encoding unique works resulting from human creativity and innovation, i.e., what intellectual property rights generally protect.Thanks to the *Deontic Expression* object representation, we can create referable rights and duties and save the association between this reference and the relevant party directly in the ledger in an immutable way through NFTs.

IV. IMPLEMENTATION

In the following, we present a possible set of technologies that enable the implementation of an SCM.

A. MPEG-21 CEL/MCO Contracts Parser

Generally speaking, the input for generating an SCM would consist of the text of a narrative contract. However, this is not in the scope of our work, and thus, we only consider a fully validated MPEG-21 CEL/MCO contract as input. For CEL, it means an XML document, while for MCO can be an RDF/TURTLE encoded file. In both cases, we have implemented a system with two different parser components that generate a set of Media Contractual Objects as described in the previous section. These Media Contractual Objects are used to generate an SCM.

B. Key Pair

The key pair is at the core of the identity authentication for contract parties. Using a digital signature as a binding cryptographic method enables any party to be represented by its public key since, by signing a digital document using the associated private key, anyone can verify that the signature is associated with the key pair's public key. In Ethereum, the public key is then transformed into an address. Thus *Party* objects can be mapped uniquely to Ethereum address. Conversely, the private key is stored (and protected) in the local device used by the party and used to digitally sign.

C. Ethereum and NFT

Our implementation is based on the Ethereum blockchain and its smart contracts. The Ethereum protocol allows smart contracts to "talk" between each other directly on-chain and to operate cryptocurrencies flow, i.e., coins and tokens. Our SCM implementation involves using an NFT smart contract, i.e., the ERC721 Non Fungible Token, that can be considered a

This article has been accepted for publication in IEEE MultiMedia. This is the author's version which has not been fully edited and
content may change prior to final publication. Citation information: DOI 10.1109/MMUL.2023.3303393

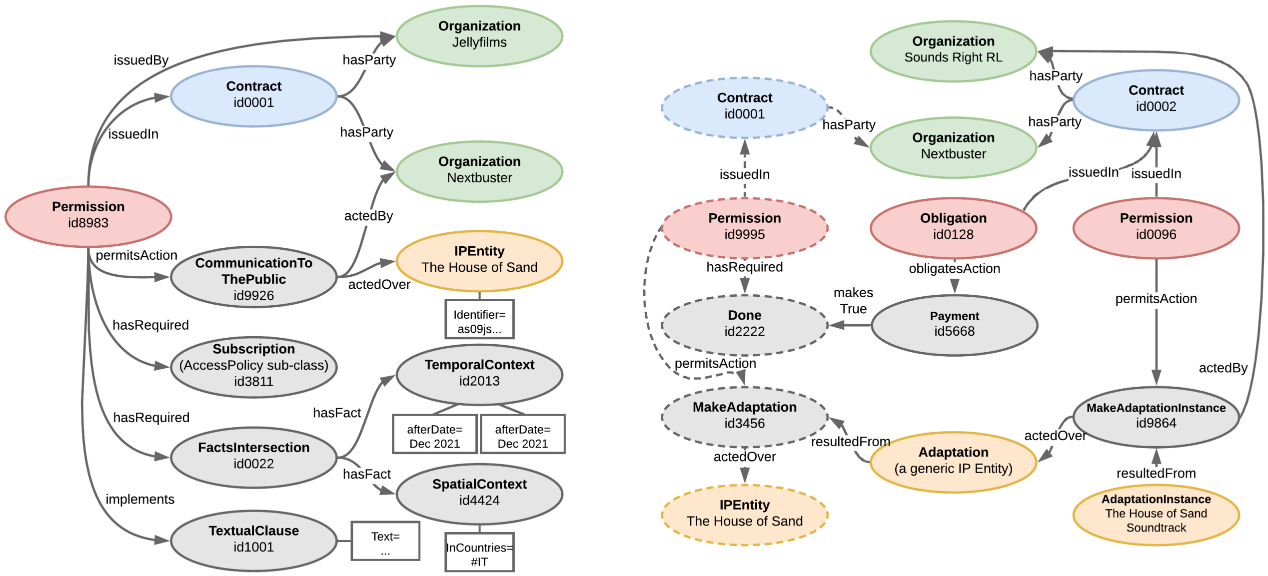

Fig. 2. Contracts for the communication to the public and the making of an adaptation instance represented using the MPEG-21 MCO. Ovals represent objects; arrows represent objects' properties and can link to other objects or data values in boxes. Objects in red represent *Deontic Expression*, in blue *Contract*, in green *Party*, in yellow *IP Entity*; the others, in gray, represent *Facts* or *Actions*.

registry for enumerating tokens (see Figure 3). The generation of a new NFT is based on such a registry: (i) when a new token is created, the ERC721 smart contracts stores in its registry a binding between the owner's Ethereum address a the token id, i.e., a unique alphanumeric identifier; (ii) the registry associates then the token id to some metadata; (iii) the metadata identifies an element described by Media Contractual Object, e.g., *IP Entity* or *Deontic Expression*; (iv) the information describing such Media Contractual Objects is stored in an off-chain document (this will be better defined later).

Using a single structure representation on-chain, i.e., the registry, enables interoperability at the smart contract level. Moreover, the use of the ERC721 NFT enables interoperability between web platforms that already implement software interfaces with the Ethereum blockchain.

D. Immutable URIs and InterPlanetary File System

In order to be immutable, data do not necessarily have to be stored directly on-chain. Indeed, this practice is costly in terms of time, space, and economics in most DLTs. Off-chain data storage, accompanied by on-chain storage of the result of the cryptographic hash function, is a trade-off that allows for information immutability and more cost-effective processing. It is also in line with data protection requirements and non-disclosure agreements. Each Media Contractual Object is stored in documents that can be maintained private, while their hash, and thus their immutability verification, can be made public. Several off-chain storage solutions can be used to store the documents, from traditional centrally-managed relational databases to distributed ones and Decentralized File Systems (DFS). In our implementation, we store contents on the InterPlanetary File System (IPFS), thatstores and shares files and directories in the form of IPFS objects identified by a CID (Content IDentifier), i.e., an immutable URI.

V. Use Case and Performance Evaluation

This section describes a use case where independent producers benefit from the proposed solution in a Video-On-Demand Services scenario.

A. Use Case

Our use case is in line with current spread of video-on-demand (VOD) services. In compliance with the European Union's Audiovisual Media Services Directive, some member states have adopted laws to accommodate the new audiovisual environment by forcing foreign VOD services to contribute to the production of domestic audiovisual content. Measures may take the form of financial contributions through film levy or obligations to invest directly in content [15].

1) Scenario: A VOD service called Nextbuster is willing to reach an agreement with an independent producer, Jellyfilms, for the co-production and distribution of a TV series entitled "The House of Sand". The contract stipulated includes an agreement for the acquisition, subject to the co-production agreement, by Nextbuster of 100% of the exploitation rights of the TV series by Jellyfilms by subscription as a form of payment in Italy for 5 years since December 2021. Furthermore, the agreement also includes a 45% share of the revenues from the exploitation of secondary rights in favor of Jellyfilms, e.g., from the sale of the soundtrack and merchandise. The latter is achieved in a second contract that Nextbuster stipulates with the Sounds Right Record Label for the production of the "The House of Sand Soundtrack".

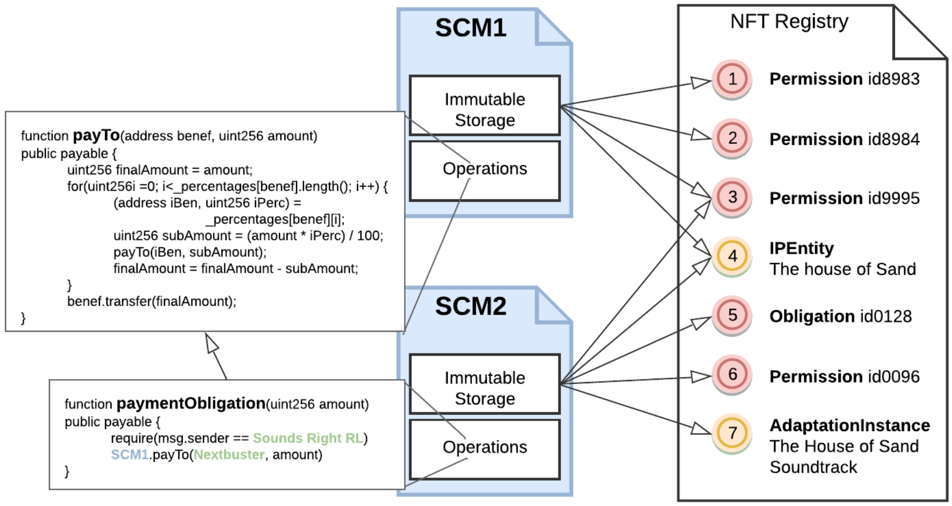

Fig. 3. Use Case Smart Contracts for Media

2) Media Contracts: We created media contracts using the MPEG-21 MCO specification based on the scenario's agreements. We now describe these with the aid of Figure 2. The *Contract* object with id=*id0001* represents the information extracted from the narrative contract between Nextbuster and Jellyfilms, i.e., the first contract. Such *Contract* object is linked with two *Permission* objects in our example but can be linked with many more objects. The first permission is a *Deontic Expression* object that permits the action *CommunicationToThePublic* by Nextbuster over the *IP Entity* object "The House of Sand", meaning that the VOD service can provide the streaming of such TV series, but under some restrictions. Such requirements are that (i) the access policy to the TV series must be by a subscription payment, (ii) that is valid from Dec 2021 to Dec 2026, and (iii) that is valid only in Italy. The *Permission* object also includes the narrative text clause. Furthermore, the second *Permission* object allows Nextbuster to trade the permission to make an adaptation of the TV series and sets an income percentage of 45% for Jellyfilms. The second contract, i.e., the *Contract* object with id=*id0002*, represents the information extracted from the narrative contract between Nextbuster and Sounds Right RL. It includes a new *Permission* object, a new *Obligation*, and a reference to one *Permission* of the first contract. The new *Permission* object issued in the second contract, indeed, permits to make of an adaptation instance to Sounds Rights RL with the obligation to make a payment in favor of Nextbuster. This *Payment* action makes true a requirement of the *Permission* for making an adaptation of the first contract, i.e., it enacts the trade defined in the first contract and thus enacts also the income percentage clause. The result of this, i.e., the adaptation instance, is the production of "The House of Sand Soundtrack".

B. Smart Contracts Performance Evaluation

We implemented the Smart Contracts Templates and software components to convert the Media Contract into a Smart Contract for Media. The implemented smart contracts have been developed in Solidity and then stored as Open Source code on Zenodo [16]. We refer the reader to our prior work where IPFS was tested out [17]. Figure 3 shows a graphical representation of the two smart contracts. In Ethereum, *gas* is a unit that measures the amount of computational effort that takes to execute operations in Ethereum smart contracts. We use gas to evaluate the performances. In Figure 4, we provide the maximum number of operations per second in the Ethereum public blockchain. The operation for minting new NFTs for representing *Deontic Expression* or *Object* objects uses $82k$ gas units on average. For the setup of the data structures, gas usage is higher. In the first minting in the NFT registry the operation uses $112k$ gas units while the first minting for a new address (or party) is $97k$. In our use case, the minted NFTs are 7 in total and thus the total amount of gas used would be around 574k. All these NFTs are referenced in the smart contracts implementing the two contracts through their id (as in Figure 3). Deploying the smart contract that represents the *Contract* object with id=id0001 uses around $2,800k$ gas units. This gas usage can be lowered by using smart contract templates that deploy optimization techniques, such as Factory or Proxy patterns. This smart contract implements the method *payTo()* (see Figure 3), which is used to subdivide the payments to an actor on the basis of the royalties agreement. In our use case, the payment of Sounds Right RL in favor of Nextbuster is split with the Jellyfilms party with an income percentage of 45%. The execution of this method uses $44k$ gas units on average.

VI. Analysis

The joint use of Semantic Web and DLTs technologies, in this case of rights management in the Video-on-Demand sector, has yielded the following advantages.

- Interoperability. The use of Semantic Web technologies automatically connects information represented us-

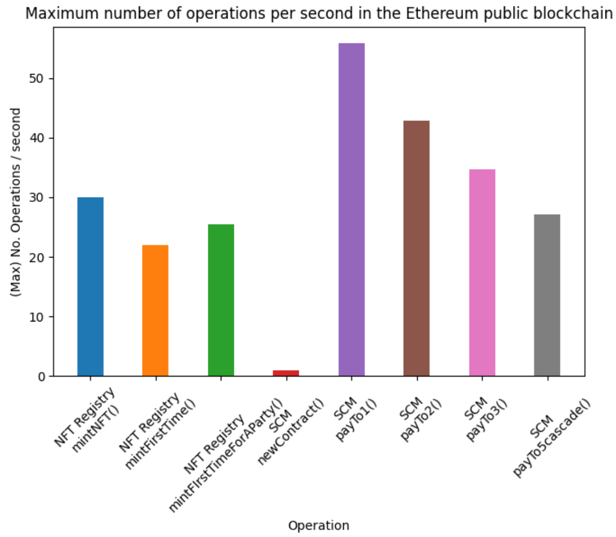

Fig. 4. Maximum number of operations per second in the Ethereum public blockchain. These results were obtained considering that a maximum of $30,000k$ gas units can be executed in an Ethereum block, so for each operation we divide this number by the gas usage required to execute such an operation. Then, we divide the obtained value by 12.1, which is the seconds, on average, between the issuance of one block and the next in the public blockchain.

ing MPEG-21 MCO with other existing ontologies and knowledge bases.

- Reasoning capabilities. A reasoner software can derive new facts from existing ones. For example, the agreement limited to customers in Italy can be easily extended with a subsequent agreement in Lazio region, i.e., by inferring that Lazio is in Italy.
- Transparency. The rights management can be visible to all the parties, including the weaker ones, such as the independent producer Jellyfilms in the presented use case.
- Technology neutrality. Using international standards and the ability to enforce the contracts without additional proprietary-based software reduces complexity and makes the solution technologically neutral.
- Better compliance. The automated execution of contracts reduces the chances of paying penalties for contract violations, reducing risks to companies before legal suits.
- Data governance. These technologies support the implementation of data governance within Common European Data Spaces, in particular in the Media Data Space, which aims at enabling transparency and control in the exchange of media assets. Our solution directly addresses *Digital Sovereignty: Enforcing data usage rights*, and indirectly *sharing by design*, *decentralisation*, *veracity* and *security*.

With regards to our implementation, results show that our smart contracts execution cost is in line with state-of-the-art solutions. The gas usage is in line with other NFT registry implementations in Ethereum, e.g., the ones that are normally used for NFT art or profile pictures [3]. In general, the increase in gas usage follows linearly the number of parties involved in the payment in a formula such as $f = 30k + (14k * beneficiaries\_n)$.

VII. Conclusions

This paper has demonstrated how to use the Smarts Contracts for Media specification, a proposed bridge to transform rights information and contracts from its MPEG-21 CEL/MCO form to smart contracts executable in blockchains. Details on the implementation have been given, and a specific use case for Video-on-Demand services has been examined in more detail. In future works we aim to develop new standard-based applications, public contract templates that can be quickly replicated and personalized, and cookbook recipes to leverage the reasoning capabilities or mappings to other ontologies and knowledge graphs. These developments will contribute towards the ecosystem of MPEG-21 SCM resources.

Acknowledgments

This work has received funding from the EU H2020 research and innovation programme under the MSCA ITN European Joint Doctorate grant agreement No 814177 LAST-JD-RIoE. The authors of this work contributed to the editing of the ISO/IEC IS 21000-23 Part 23.

References

[1] P. Kudumakis, T. Wilmering, M. Sandler, V. Rodríguez-Doncel, L. Boch, and J. Delgado, "The challenge: From mpeg intellectual property rights ontologies to smart contracts and blockchains [standards in a nutshell]," *IEEE Signal Processing Magazine*, vol. 37, no. 2, pp. 89–95, 2020.

[2] A. Savelyev, "Copyright in the blockchain era: Promises and challenges," *Computer law & security review*, vol. 34, no. 3, pp. 550–561, 2018.

[3] S. Casale-Brunet, M. Zichichi, L. Hutchinson, M. Mattavelli, and S. Ferretti, "The impact of nft profile pictures within social network communities," in *Proc. of the Conference on Information Technology for Social Good (GoodIT)*. ACM, 2022, pp. 1–11.

[4] T. Pellegrini, A. Schönhofer, S. Kirrane, S. Steyskal, A. Fensel, O. Panasiuk, V. Mireles-Chavez, T. Thurner, M. Dörfler, and A. Polleres, "A genealogy and classification of rights expression languages-preliminary results," in *Data Protection/LegalTech-Proceedings of the 21st International Legal Informatics Symposium IRIS*, 2018, pp. 243–250.

[5] V. Rodriguez-Doncel and J. Delgado, "A media value chain ontology for mpeg-21," *IEEE MultiMedia*, vol. 16, no. 4, pp. 44–51, 2009.

[6] V. Rodriguez-Doncel, J. Delgado, S. Llorente, E. Rodriguez, and L. Boch, "Overview of the mpeg-21 media contract ontology," *Semantic Web*, vol. 7, no. 3, pp. 311–332, 2016.

[7] E. Rodríguez, J. Delgado, L. Boch, and V. Rodríguez-Doncel, "Media contract formalization using a standardized contract expression language," *IEEE multimedia*, vol. 22, no. 2, pp. 64–74, 2014.

[8] W.-S. Lee, J. A, H.-C. Hsu, and P.-A. Hsiung, "Spchain: A smart and private blockchain-enabled framework for combining gdpr-compliant digital assets management with ai models," *IEEE Access*, vol. 10, pp. 130 424–130 443, 2022.

[9] W.-S. Lee, J. A., H.-C. Hsu, and P.-A. Hsiung, "A smart and private blockchain-enabled framework for digital assets," in *Proceedings of the 2022 5th International Conference on Blockchain Technology and Applications*, ser. ICBTA '22. New York, NY, USA: Association for Computing Machinery, 2023, p. 34–39. [Online]. Available: https://doi.org/10.1145/3581971.3581976

[10] A. Ayub Khan, A. A. Wagan, A. A. Laghari, A. R. Gilal, I. A. Aziz, and B. A. Talpur, "Biomt: A state-of-the-art consortium serverless network architecture for healthcare system using blockchain smart contracts," *IEEE Access*, vol. 10, pp. 78 887–78 898, 2022.

[11] D. Chalmers, R. Matthews, and A. Hyslop, "Blockchain as an external enabler of new venture ideas: Digital entrepreneurs and the disintermediation of the global music industry," *Journal of Business Research*, vol. 125, pp. 577–591, 2021.

[12] R. Ianella, "Open digital rights language (odrl)," *Open Content Licensing: Cultivating the Creative Commons*, 2007.
[13] A. Banerjee and K. P. Joshi, "Link before you share: Managing privacy policies through blockchain," in *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017, pp. 4438–4447.
[14] T. Gaber, A. Ahmed, and A. Mostafa, "Privdrm: A privacy-preserving secure digital right management system," in *Proceedings of the Evaluation and Assessment in Software Engineering*, 2020, pp. 481–486.
[15] I. Kostovska, T. Raats, and K. Donders, "The rise of the 'netflix tax'and what it means for sustaining european audiovisual markets," *Innovation: The European Journal of Social Science Research*, vol. 33, no. 4, pp. 423–441, 2020.
[16] M. Zichichi, "miker83z/MPEG-21-Part-23-DLT-Manager: mpeg-21-part-23-dlt-manager-v1.0.0," Nov. 2022. [Online]. Available: https://doi.org/10.5281/zenodo.7327351
[17] M. Zichichi, S. Ferretti, G. D'Angelo, and V. Rodríguez-Doncel, "Data governance through a multi-dlt architecture in view of the gdpr," *Cluster Computing*, pp. 1–28, 2022.