Smart Contracts Taxonomy ver 1.0

Status	VER 1.0
Outcome	Smart Contracts Taxonomy

Google Doc Link

DOC : Unknown macro: 'lref-gdrive-file'

PDF : Unknown macro: 'lref-gdrive-file'

Background

- Determine a suitable structure for the taxonomy
- Capture the relationships inherent in the information
- Reflect how the information fits into smart contracts usage and scenarios

Mind map

Unknown macro: 'gliffy'

Smart Contracts Taxonomy

	Functional Requirements	Computation	Law
	Privacy and Confidantiality • Differential Privacy • Secure Multiparty Computation	Smart Contract Languages • Turing Complete • Turing Incomplete	Regulation

	Computational Characteristics	Virtual Machines and Runtimes
		• ABIs
		• WASM
		  ◦ External Function Calls
			Statute
			• Dispute Resolution
			  ◦ Case Law

	Interaction • Human • Machine-to-Machine	Formal Methods • Halting	Constitution
	Packaging • Versioning ◦ Upgradeable	Governance • Dispute Resolution	Governance
	AI Driven • Oracles		International Treaties
	Interoperability • Between Smart Contracts • Between Engines		
Relates	Functional Computation	Law Dispute Resolution	Computation Governance

Smart Contracts Taxonomy Definitions

Privacy and Confidentiality:

Privacy is the individual's right to keep their data to themselves and not to have their actions recorded or monitored. Confidentiality is about controlling who has access to sensitive information.

Interaction:

Smart contracts can call functions of other contracts and are even able create and deploy other contracts.

Packaging – Versioning-Upgrading:

Due to the immutable nature of the blockchain, it's not possible to change the code of a deployed smart contract once it has been deployed.

AI Driven:

Enabling AI on the blockchain and integrating AI to be nested within the smart contract offers us a powerful solution. Blockchains and smart contracts cannot access data from outside of their network. In order to know what to do, a smart contract often needs access to in- formation from the outside world that is relevant to the contractual agreement, in the form of electronic data, also referred to as oracles. These oracles are services that send and verify real world occurrences and submit this information to smart contracts, triggering state changes on the blockchain.

Interoperability:

On a surface level, interoperability allows for the information of blockchain A and blockchain B to interact with each other through systems that are complimentary on either blockchain. Typically this is in the form of compatible smart contracts with partial keys on either chain or via oracles.

Languages:

Almost all popular programming languages are used in the blockchain industry, however developers have to consider what type of development they would like to undertake as different languages are used for certain blockchain projects and applications.

Virtual Machines And Runtimes:

The Ethereum Virtual Machine or EVM is the runtime environment for smart contracts in Ethereum. It is not only sandboxed but actually completely isolated, which means that code running inside the EVM has no access to network, filesystem or other processes. Smart contracts even have limited access to other smart contracts.

Formal Methods - Halting:

The most popular smart contract languages favor expressiveness rather than safety, and bugs in smart contracts have already lead to significant financial losses from accidents. Smart contracts are also appealing targets for hackers since they can be monetized. For these reasons, smart contracts are an appealing opportunity for systematic auditing and validation, and formal methods in particular. The halting problem was first mentioned by Alan Turing and it states that it is impossible to build an algorithm capable decide if a program will terminate its execution for all possible inputs.

Governance – Dispute Resolution:

Blockchain governance depends on the interactions of node holders, token holders, and core developers. Smart contracts can be used to manage the agreements made between these parties. In blockchain terms, this is known as a decentralized autonomous organization (DAO). The two biggest legal problems of smart contracts lie in their blockchain provenance. The first one is the enforceability of smart contracts. There is also the problem of jurisdiction. How will disputes involving smart contracts for international transactions that span multiple geographies be resolved?

Law Regulation:

The legality of smart contracts is often ambiguous—regulations need to be clarified, and a smart dispute-resolution system must be created. Whether or not a smart contract can act as a legally enforceable contract depends on the specifics of each case.

Statute – Dispute Resolution:

See Governance – Dispute Resolution above.

Constitution

Smart contracts' operation and place in existing contract law. There is a distinction between strong and weak smart contracts, as defined by the costs of their revocation and modification. Smart contracts are simply a new form of preemptive self-help that should not be discouraged by the legislatures or courts.

Governance

See Governance – Dispute Resolution above.

International Treaties

Can the blockchain be used as a licensing tool for 'international' copyright rights? Although it is not inaccurate to speak of international copyright law, as contained for example in international treaties, there is no such thing as an international copyright right. The treaties recognize the protection of copyright in multiple jurisdictions, but based on the law of each jurisdiction.

About the Hyperledger Smart Contracts Working Group

The main goal of this workgroup is to give an academic perspective to the smart contracts research topics and in parallel make clear to users, developers, researchers, businessmen, decision makers and others interested in smart contracts practical ways to utilize them on the different DLTs that are under the Hyperledger umbrella and explore all potentials from deploying them in everyday software solution scenarios. The scope is to define concepts regarding smart contracts and to produce material to describe the various aspects and meanings, trying to come up to standards or good practices. The audience for smart contracts is large and spans from researchers, developers, businessmen, decision makers, policy makers, law makers, software users, citizens to governments, banks, financial institutions, insurance providers, etc

Two main research topics and separation of interests are:

- **Technology oriented**
- Law oriented

To learn more about this working group and find out how to participate, visit:

https://wiki.hyperledger.org/display/SCWG/Smart+Contracts+Working+Group

References

1. https://hackernoon.com/smart-contracts-privacy-vs-confidentiality-645b6e9c6e5a
2. https://zupzup.org/smart-contract-interaction/
3. https://hackernoon.com/how-cortex-brings-ai-on-the-blockchain-86d08922bb2a
4. https://blockchainhub.net/blockchain-oracles/
5. https://fitznerblockchain.consulting/interoperability-the-key-to-unlocking-the-potential-of-blockchain/
6. https://solidity.readthedocs.io/en/v0.5.10/introduction-to-smart-contracts.html
7. https://link.springer.com/chapter/10.1007/978-3-030-03427-6_22
8. https://medium.com/aelfblockchain/running-blockchain-governance-with-smart-contracts-954fa0c3c747
9. https://www.investopedia.com/news/how-are-disputes-smart-contracts-resolved/
10. https://hackernoon.com/smart-contracts-part-2-the-legality-761cc4be100d
11. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2959166
12. https://academic.oup.com/ijlit/article/26/4/311/5106727