Explainer | Dual Integration

Jump to:

Quick Links:

Explainer | Ecosystem Applications (https://monax.io/explainers/ecosystem_applications/)

Explainer | Blockchains (https://monax.io/explainers/blockchains/)

Explainer | Permissioned Blockchains (https://monax.io/explainers/permissioned_blockchains/)

Explainer | Smart Contracts (https://monax.io/explainers/smart_contracts/)

Explainer | Smart Contracts v. Tokenized Approaches (https://monax.io/explainers/contracts_v_tokens/)

Explainer | Legal Engineering (https://monax.io/explainers/legal_engineering/)

Putting the Contracts in Smart Contracts

Smart contracts are useful for automating many data-driven relationships. However, to be enforceable, smart contracts must work in existing legal frameworks where those frameworks already exist, removing a degree of uncertainty in enforcement. In order to bridge the gap between existing electronic contracts law and blockchain smart contracts, Monax offers *dual integration*. For more on smart contracts, see our explainer (https://monax.io/explainers/smart_contracts/).

Dual integration is the process of integrating a specific legal contract into a specific smart contract running on a distributed data store, such as Monax. This allows parties to use established dispute resolution processes in the jurisdiction(s) of choice while also using a smart contract as the primary mechanism for administering the data-driven interaction that attends to the agreement between the parties.

Dual integration enables legal enforceability of smart contracts

The rationale behind dual integration is that for the forseeable future, legal systems are unlikely to resolve disputes stemming from smart contracts solely on the basis of their code. Courts will apply the defaults for similar prose agreements, an end that may not reflect the intention of the smart contracting parties. Because of the risks involved in

enforcing smart contracts by code alone, we highly encourage all smart contract systems developers to utilize dual integration of some kind. Dual integration means that a smart contract will be linked to a document that can be enforced by a court.

Basic dual integration:

1. Deploy a smart contract that is capable of storing a checksum

2. Reference the chainId and contractAddress of the deployed smart contract in the final draft of the real world contract.

3. Execute the prose contract and produce its checksum (AKA "hash" or "digital fingerprint")

4. Send a transaction storing the checksum of the prose contract into the smart contract.

That's it. If those steps are followed then there will be a cryptographically-certain dual integration.

Once the dual integration process is finished the result will be a smart contract which references a specific fingerprint of a real world contract (this is called the hash of a file, or sometimes it is called the checksum of a file) and a real world contract which integrates the specific fingerprint of a smart contract (this utilizes the chainID of the distributed data store which the smart contract runs on as well as the contractAddress of the specific smart contract). This circular reference ensures – to a cryptographic certainty – that the specific file which contains the real world contract and the specific smart contract both properly reference one another.

Monax has created a dual integration tool (https://github.com/monax/hello-doug/tree/master/deprecated_apps/dual_integration) to get smart contract writers going.

◀ All Explainers (https://monax.io/explainers/)

Copyright © 2014-2017 Monax