Accord Project ID: The Smart Legal Contract Identity and Trust Framework Standard
Overview

Accord Project and Smart Legal Contracts

Identity Aspects of Smart Legal Contracts

Decentralized Identity for Smart Legal Contracts

Identity Trust Framework for Smart Legal Contracts
What is the Accord Project?

Sets standards for **smart legal contracts** by interfacing with leading lawyers, industry organizations, and technologists.

Addresses the lack of standards for smart legal contracts and the widely divergent, potentially incompatible, approaches that are emerging.

Producing an **open source middleware core** for smart legal contracts that embodies the techno-legal standards and meets the needs of the legal industry.

---

**IN COLLABORATION WITH**

- Clio
- IEEE
- IACC
- ISO
- OIX OPEN IDENTITY EXCHANGE
## Contract Purposes

<table>
<thead>
<tr>
<th>Ranking</th>
<th>To what extent should a contract fulfil the following purposes?</th>
<th>% stating ‘to a high extent’</th>
<th>Overall score (scale 1-5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>A record of rights, responsibilities and obligations</td>
<td>76</td>
<td>4.71</td>
</tr>
<tr>
<td>2</td>
<td>Providing protection and remedies in the event of a dispute</td>
<td>65</td>
<td>4.53</td>
</tr>
<tr>
<td>3</td>
<td>A framework for a mutually successful business outcome</td>
<td>61</td>
<td>4.46</td>
</tr>
<tr>
<td>4</td>
<td>A tool for risk apportionment</td>
<td>52</td>
<td>4.31</td>
</tr>
<tr>
<td>5</td>
<td>Support for a business relationship</td>
<td>48</td>
<td>4.27</td>
</tr>
<tr>
<td>6</td>
<td>Governance and performance management</td>
<td>42</td>
<td>4.23</td>
</tr>
<tr>
<td>7</td>
<td>A tool for risk management</td>
<td>41</td>
<td>4.1</td>
</tr>
<tr>
<td>8</td>
<td>An effective communication tool for those with a need to know</td>
<td>46</td>
<td>4.07</td>
</tr>
<tr>
<td>9</td>
<td>Providing operational guidance</td>
<td>30</td>
<td>3.85</td>
</tr>
<tr>
<td>10</td>
<td>An instrument for generating financial benefit</td>
<td>26</td>
<td>3.76</td>
</tr>
<tr>
<td>11</td>
<td>Demonstrating brand and corporate values</td>
<td>12</td>
<td>3.08</td>
</tr>
</tbody>
</table>

Source: IACCM
Smart Legal Contracts
Contract Text versus Code
Lack of Techno-Legal Standards

Algorithm 1. Pseudo-code of the licensing contractual clauses.

1. Initialize getLicense, getApproval, getCommission, use, publish, comment, remove
2. If [Forb.licensee] use = true
3. [Perm.licensee] publish = true
4. [Perm.licensee] comment = true
5. violation = false
6. procedure EVALUATE_LICENSE_CONTRACT
7. if getLicense = true then
8. [Forb.licensee] use = false
9. [Perm.licensee] publish = true % Article 1
10. if getLicense = true and (getApproval = true or getCommission = true) then
11. [Perm.licensee] publish = false
12. [Perm.licensee] comment = true
13. violation = false % Article 2, 4
14. if getLicense = true and
15. getApproval = false and
16. getCommission = false and
17. publish = true then
18. [Obl.licensee] remove = true % Article 2
19. if [Perm.licensee] publish = true then
20. [Forb.licensee] use = false
21. [Perm.licensee] comment = true % Article 3
22. if getLicense = true and getCommission = true then
23. [Perm.licensee] publish = false
24. [Obl.licensee] publish = true
25. [Perm.licensee] comment = true
26. violation = true % Article 4
27. if [Forb.licensee] use = true and [use = true] or
28. [Perm.licensee] publish = true and [publish = true] or
29. [Obl.licensee] publish = true and [comment = true] or
30. [Perm.licensee] comment = true and [comment = true] or
31. violation = true % Article 5
32. if violation = true then
33. [Forb.licensee] use = true
34. [Perm.licensee] publish = true
35. [Perm.licensee] comment = true
36. [Perm.licensee] use = false
37. [Perm.licensee] publish = false
38. [Perm.licensee] comment = false
39. [Obl.licensee] publish = false

Article1.0: => [Forb_licensee] use
Article1.1: getLicense => [Perm_licensee] use
Article2.1: => [Forb_licensee] publish [Compensated] [Obl_licensee] remove
Article2.2: getApproval => [Perm_licensee] publish
Article3.1: => [Forb_licensee] comment
Article3.2: [Perm_licensee] publish => [Perm_licensee] comment
Article4.1: getCommission => [Obl_licensee] publish
Article4.2: getCommission => getLicense
Article5: violation => [Forb_licensee] use
% Superiority relation
Article1.1 > Article1.0, Article8 > Article1.1,
Article2.2 > Article2.1, Article3.2 > Article3.1

Source: Idelberger, Governatori, Riveret, Sartor (2016)
Accord Project: Smart Legal Contract Templating

- **Execution Verification**
- **Editing Validation**
- **Logic**
- **Natural Language**

**Template**
Smart *Legal* Contracts vs. Smart Contracts

“A smart contract is an automatable and enforceable *agreement*. Automatable by computer, although some parts may require human input and control. Enforceable either by legal enforcement of rights and obligations or via *tamper-proof* *execution* of computer code.”

- Clack, Bakshi, Braine, 2016

“Smart contracts are code that is stored and executed on a blockchain. Add a user interface and smart contracts serve as the backends for decentralized applications, or dapps.”

- Mike Goldin, ConsenSys
Smart Legal Contract and DLT

Blockchain light

Blockchain heavy

Source: Idelberger, Governatori, Riveret, Sartor (2016)
Case Study: Perishable Goods Contract

Participants: grower (farmer), shipper, importer, port authority, finance and insurance

Contractual provisions:

• quantity be shipped in containers with sensor readings of a certain frequency

• be shipped under temperature conditions within a certain range as indicated by sensor readings

• be shipped under humidity conditions within a certain range as indicated by sensor reading

• penalty for violation of temperature or humidity conditions
@AccordTemplateModel("perishable-goods")
concept TemplateModel {
    --> Grower grower
    --> Importer importer
    --> Shipment shipment
    o DateTime dueDate
    o Double unitPrice
    o Unit unit
    o Integer minUnits
    o Integer maxUnits
    o String product
    o Integer temperatureReadingFrequency
    o Duration duration
    o Double minTemperature
    o Double maxTemperature
    o Double minPenaltyFactor
    o Double maxPenaltyFactor
}

/** * Execute the smart clause
 * @param {Context} context - the Accord context
 * @param {org.accordproject.perishablegoods.ShipmentReceived} context.request - the incoming request
 * @param {org.accordproject.perishablegoods.PriceCalculation} context.response - the response
 * @AccordClauseLogic
 */
function payOut(context) {
    logger.info(context);
    var shipmentReceived = context.request;
    var shipment = shipmentReceived.shipment;
    var res = context.response;
    res.shipment = shipment;
    var data = context.data;
    var payOut = data.unitPrice * shipmentReceived.unitCount;
    logger.info('Base payOut: ' + payOut);
    logger.info('Received at: ' + shipmentReceived.timestamp);
    logger.info('Contract arrivalDateTime: ' + data.dueDate);
    if(shipmentReceived.unitCount < data.minUnits || shipmentReceived.unitCount > data.maxUnits) {
        throw new Error('Units received out of range for the contract. ');
    }
}
Smart Legal Contract and DLT Example

Sensor Data from IoT Device

Distributed Ledger

perishable-goods

On receipt of the shipment "SHIP_001" the importer "DAN" pays the grower "PETER" 1.50 USD per KG. The shipment must contain between 3000 and 3500 KG of "Grade 1, Size 4, Zutano Mexican Avocados". Shipping containers used must be temperature controlled, and...

Version: 0.0.2

Accounting System
Smart Legal Contract State on Dist. Ledger

Participant registry for org.accordproject.perishablegoods.Grower

ID | Data
---|---
farmer@email.com

- `email`: "farmer@email.com"
- `address`: {
  - `street`: "org.accordproject.perishablegoods.Address",
  - `country": "USA"
}
- `accountBalance": 4500

Shipment data on DL
Sensor data on DL
Participant data on DL
Comments
Background on Identity

Identity: what something is, associated characteristics, and capabilities

Identity transactions

Players: identity providers, verifiers, subjects, relying parties

Verifiable claim: statement about a subject relating its credentials such as qualification, characteristics, achievement, or quality
Identity Aspects of Smart Legal Contracts

Documents

Parties

Things

Computation

→ identified by identifiers
Identity of Contracting Documents

agreements, sections, clauses

invoices, statements of works, notices

type of agreement and other document
Identity of Contracting Parties and Things

- known to each other
- regulatory requirements
- persons and things involved in performance
  → trustless contracting
Identity of Contracting Computation

valid source of data

storage

compatible software system

distributed ledger
Comments
Benefits of Decentralized Identity Systems

- enabling users with (greater) control over the use of their identity
- not relying on one approach, technology, or identity provider that may be a single point-of-failure or suboptimal
- ability to employ a greater diversity of approaches and technology
- enhanced security
Decentralized Identity for Smart Legal Contracts

Decentralized identifiers (DIDs) and associated metadata

globally unique identifiers for decentralized systems

persistent - assigned once to an entity

globally resolvable (interoperable)
cryptographic verification of the identifier owner
DIDs Well-Suited for Smart Legal Contract Identity

numerous decentralized entities

entities potentially needing the ability to validate contract identifier depending on context

different contracts (or clauses) being authorized to perform specific software services depending on context

performance of contract obligations verified by various parties
Smart Legal Contract Verifiable Claims

contracts establish relationships between, and qualities about, parties and things

contract clauses often contain a wide variety of legally binding assertions about the characteristics of parties

contract rights and obligations are a type of quality or achievement

Example: port authority issues a credential to a seller that certain goods have been delivered, and a data sensor verifies the credential which entitles seller to payment
SCLCs may initiate contract operations on external systems by reference to a service endpoint - web service to invoke the SLC

Example: update contract state on blockchain

DIDs use service endpoints to initiate trusted interactions
DIDs and associated data may be read from a distributed ledger

as to SLC claims, distributed ledger may be used to

register the issuance of verifiable

verify or revoke the claim
Comments
Smart Legal Contract Identity Trust Framework

APID trust framework for smart legal contracts seeks to:

decrease the cost and risk associated with commercial contracting

increase the use and reliability of smart legal contracts

increase trust that parties have in the ability of their contract counterparties to fulfill their contract obligations and stay within the bounds of their contract rights
Smart Legal Contract Identity Trust Framework

increase trust that parties have in the ability of their contract counterparties to provide remedies in case breach

reduce the amount of information parties are required to obtain about counterparties to be comfortable doing business with them

increase parties’ willingness to rely on automated contract execution, operations, and other processes support the use of decentralized identity architectures such as the Sovrin Network
increase trust that parties have in the ability of their contract counterparties to provide remedies in case of breach

reduce the amount of information parties are required to obtain about counterparties to be comfortable doing business with them

increase parties’ willingness to rely on automated contract execution, operations, and other processes
Comments