LEGALLY BINDING APPLICATIONS WILL DEFINE FUTURE BLOCKCHAINS

Presentation · May 2018
DOI: 10.13140/RG.2.2.21340.08323

CITATIONS
0

3 authors, including:

Volker Skwarek
Hochschule für Angewandte Wissenschaften Hamburg
52 PUBLICATIONS · 66 CITATIONS

Some of the authors of this publication are also working on these related projects:

- Swarm-based Low Power Localization View project
- Smart Contracts on Blockchains View project

All content following this page was uploaded by Volker Skwarek on 04 June 2018.

The user has requested enhancement of the downloaded file.
LEGALLY BINDING APPLICATIONS WILL DEFINE FUTURE BLOCKCHAINS

DLT Reg

Prof. Dr.-Ing. Volker Skwarek
30.05.2018
TWO ASPECTS OF BLOCKCHAINS AND LAW WE SHOULD TALK ABOUT

a) blockchain → world:
How can blockchains **support legal workflows** in the physical world (= **smart contracts**)?

b) world → blockchain:
How do blockchains **legally fit** into the real world (= **legal compliance**)?

c) world ↔ blockchain:
The future of the world with and of blockchains!
TWO ASPECTS OF BLOCKCHAINS AND LAW WE SHOULD TALK ABOUT

a) blockchain → world:
How can blockchains **support legal workflows** in the physical world (= **smart contracts**)?

b) world → blockchain:
How do blockchains **legally fit** into the real world (= **legal compliance**)?

c) world ↔ blockchain:
The future of the world with and of blockchains!
WHY DO WE TALK ABOUT CONTRACTS ON BLOCKCHAINS
„NICK SZABO“

Usual quotation of Nick Szabo („Idea of Smart Contracts“, 1997, paragraph 2 center):

“A canonical real-life example, which we might consider to be the primitive ancestor of smart contracts, is the humble vending machine. Within a limited amount of potential loss (the amount in the till should be less than the cost of breaching the mechanism), the machine takes in coins, and via a simple mechanism, which makes a freshman computer science problem in design with finite automata, dispense change and product according to the displayed price.

[…]

Smart contracts go beyond the vending machine in proposing to embed contracts in all sorts of property that is valuable and controlled by digital means.”
Same quotation from Nick Szabo („Idea of Smart Contracts“, 1997, paragraph 2, Beginning):

“Many kinds of **contractual clauses** (such as collateral, bonding, delineation of property rights, etc.) can be embedded in the hardware and software […]”

**Important addition:** Nick Szabo talks about **contractual clauses = contractual elements**, not necessarily only about **contracts**
different views on smart contracts

- **technically**: Smart Contracts are *distributed applications*, also used for automated execution of contract clauses
- **legally**: whether or not a distributed application named smart contract has a legally binding intention, *decides a court*.
- **consensus**: If a smart contract is defined and designed for a legal intentention, the smart contract has to be legal.
(PRELIMINARY) ISO-DEFINITION VON SMART CONTRACTS

- distributed code representing a process automation
- executed on a blockchain or distributed ledger which,
- once validated and confirmed,
- results in an outcome
- that is agreed upon by participants in a transaction.

Note to entry: The outcome of a smart contract may or may not primarily intended to be legally binding.
AUTOMATION OF COMPLEX LEGAL PROCESSES USING SMART CONTRACTS

BRAVE NEW WORLD…
TWO ASPECTS OF BLOCKCHAINS AND LAW WE SHOULD TALK ABOUT

a) blockchain → world:
How can blockchains support legal workflows in the physical world (= smart contracts)?

b) world → blockchain:
How do blockchains legally fit into the real world (= legal compliance)?

c) world ↔ blockchain:
The future of the world with and of blockchains!
A CONTRACT AS A SUM OF ITS ELEMENTS

No, it’s more than the sum of its elements! - every contract is embedded into a legal framework.
i.e. Article 2, Paragraph 1 (Basic Law for the Federal Republic of Germany = German Constitution) allows the freedom of contracting in general – but not in any kind:

„Every person shall have the right to free development of his personality insofar as he does not violate the rights of others or offend against the constitutional order or the moral law.”

A defined frame defines (and partly restricts) the freedom of
• contractual parties (e.g. above age of 14, 16, 18 years)
• content (no legal violation, no immoral contracts)
• form (free form agreement, written form required)

Not everything possible is allowed. Or: an automated smart contract may be automating more than it should do!
A lawyer usually has this in mind – but does a software developer even know this?
**WHAT ARE CONTRACTS LIKE?**
**A BIT OF LEGAL THEORY**

Abstract validity check of contracts in 3 steps:

<table>
<thead>
<tr>
<th>I. claim arose</th>
<th>II. Claim not lost</th>
<th>III. claim enforceable</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Conclusion of the contract: Agreement of the parties</td>
<td>• Rescission (fraudulent deception, unlawful threat or error)</td>
<td>• Plea of statute of limitations,</td>
</tr>
<tr>
<td>Validity of the contract:</td>
<td>• Condition subsequent</td>
<td>• Misuse of rights,</td>
</tr>
<tr>
<td>• Legal capacity,</td>
<td>• Impossibility,</td>
<td>• Right of retention,</td>
</tr>
<tr>
<td>• Defect of form,</td>
<td>• Termination,</td>
<td></td>
</tr>
<tr>
<td>• Violation of the Prohibition Act,</td>
<td>• Withdrawal,</td>
<td></td>
</tr>
<tr>
<td>• Immorality,</td>
<td>• Revocation,</td>
<td></td>
</tr>
<tr>
<td>• Condition precedent</td>
<td>• Offsetting,</td>
<td></td>
</tr>
<tr>
<td>• ...</td>
<td>• ...</td>
<td></td>
</tr>
</tbody>
</table>

"an objection is a subjective right that gives the beneficiary the possibility to prevent the enforcement of a claim of the creditor without destroying it. This is a negative design right."

Prof. Dr.-Ing. Volker Skwarek
Legally binding applications - DLTReg 2018
TRANSFERABILITY CONTRACTUAL CHECKS ↔ SOFTWARE DEVELOPMENT

<table>
<thead>
<tr>
<th>I. claim arose</th>
<th>II. Claim not lost</th>
<th>III. claim enforceable</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>positive check</strong></td>
<td><strong>mainly negative check</strong></td>
<td><strong>negative check, “objection”</strong></td>
</tr>
</tbody>
</table>

---

Software (engineering) view

- **irrelevant environment**
- **context**
- **system**
- **environment**

- **specification**

---

Legal view

I. claim arose
II. Claim not lost
III. claim enforceable

- positive check
- mainly negative check
- negative check, “objection”

---

Legally binding applications - DLTReg 2018
1. Software development is based on specifications that model a target state.
2. Certain negative states from the system context can be researched and actively queried by positive checks. But not all!
3. Interactions with the non-relevant environment from a specification point of view are negative tests in an infinite range of alternatives. In some cases there is even freedom of a single party to choose (objection). This cannot be modeled in advance as a collection of individual specifications.

**Consequence 2**

- Legal frameworks offer a variety beyond the specification modeling of contract clauses in the context of software.
- Legal frameworks offer possibilities that cannot be modeled by active positive checks in a software.
From the point of view of software development, it is **not possible to consider all positive cases** and certainly **not all negative exclusions**.

Therefore for an autonomous, fully automated smart contract, **implementing legal statements**, the **possibility of human interaction** needs to be implemented in **case of unforeseen situations or systematic errors**.
LEGAL ASPECTS MAKING IMPLEMENTATIONS ON BLOCKCHAINS EVEN MORE DIFFICULTY

- **Selfexecuting copies** of contracts are **arbitrarily distributed** among an **unknown number** of **unknown people**.
- Those **unknown entities** **assess** the contract for you, **judge und execute** it.
- Some of these entities are **located in foreign jurisdictions**.
- These entities initiate a transaction between the parties.
- This transaction **cannot be eliminated ex tunc** (= condition as if the legal transaction had never existed), but **can only be reversed** in a comprehensible manner.
- Nobody will give you a **statement about the execution** time –
  - or whether it will ever be executed at all –
  - or whether it stops during execution because it runs out of gas.
- **Real-time-statements are complicated** at all. Means: You may **claim an interrogation in time**, but it will simply **not be processed in time** and there is nobody to complain about.
MINIMUM REQUIREMENTS FOR SMART-CONTRACTS

In order to enable human interaction with self-sufficient electronic contracts, the following functions, among others, are useful

- **Pause function**: Responsiveness with inhibition of contract execution
- **Stop function**: Cancelability of a contract
- **Real-time function**: defined response times

and many more...

- such as "fuzzyness": intentionally blurred wording of clauses in order to be able to subsequently evaluate as many situations as possible and integrate them into the contract? or what about
- **the right for self-execution**: A contract establishes a claim (= right of fulfillment, claim) but not the right to independent performance (= execution)
TWO ASPECTS OF BLOCKCHAINS AND LAW WE SHOULD TALK ABOUT

a) blockchain → world:
How can blockchains support legal workflows in the physical world (= smart contracts)?

b) world → blockchain:
How do blockchains legally fit into the real world (= legal compliance)?

c) world ↔ blockchain:
The future of the world with and of blockchains!
WHAT IS BLOCKCHAIN REALLY?  
A DIFFERENT PERSPECTIVE MAY HELP

Our world is full of contradictions

• contractual freedom vs. regulatory rules
• full automation vs. human interaction
• applicable law vs. programmable logic

but they can be resolved with

• a programmable legal framework
• some different decision processes
• and an (at least) 2-layer-architecture
LEGALLY BINDING APPLICATIONS - DLTReg 2018
SUPPORT SOFTWARE DEVELOPERS WITH MODELS AND FRAMEWORKS
NOT TO REINVENT LAW!

Legal binding applications - DLTReg 2018
TECHNO-LEGAL PERSPECTIVE ON SMART CONTRACTS

SUMMARY

Legal perspective
- Smart contracts represent (few automatable) contract elements (= clauses) rather than entire contracts.
- Legal contract frameworks can probably be mapped in Smart Contracts, but not simply "just like that"...

Technical Perspective
- "Code is Law" cannot be legally represented by few lines of code. It requires a process and a complex framework.
- Aspects that make the blockchain attractive (immutability, unstoppable, decentralization...) are not legally attractive in terms of contract execution at all.

Common Perspective
- Blockchains enable meaningful applications for automated contract processing
- Contracts on block chains require special legal regulation as common law-books are mainly “asset-based”.
- Smart contracts require a software-framework, operating system stacks and standardization, to connect the technical and the legal world to a techno-legal world.

Legally binding applications - DLTReg 2018
THANK YOU FOR YOUR ATTENTION.

Ansprechpartner:
Prof. Dr.-Ing. Volker Skwarek
volker.skwarek@haw-hamburg.de
http://www.ls.haw-hamburg.de/~blockchain/