Smart Contracts – beyond code

What are they, are they legally binding and what is the potential use, towards sustainable contracts and interesting developments!

Ir Olivier Rikken MBA
Amsterdam 20-6-2018

AXVECO
Improving performance,
managing risk.

goto; conference

Curicullum Vitae – Olivier Rikken

Education:

Delft University of Technology, System Engineering, Policy Analysis and Management – Simulation of Logistics Systems (MSc degree)

Nyenrode Business University / Kellogg School of Management / Stellenbosch Business School, Effect of strategy changes on various aspects of organizations (MBA degree)

Previous work experience:

DailyFresh Logistics – Business Engineer – MT - Process improvement and IT

Atos Origin – Executive Business Consultant – Thought Leader BPM

GE Capital – Operations leader, Managing Sourcing, Facilities, Project Bureau (all projects e.g. SEPA) and Operational Excellence departments, further responsible for Business Continuity Management and Records Management

Current positions:

AXVECO – Director Blockchain & Smart Contracts – AXVECO is a leading consulting firm in the Netherlands on sustainable innovation specialized in blockchain awareness, consulting and implementations

Dutch Blockchain Coalition – Founder and Chairman of the Smart Contract workgroup and HCA workgroup Core team. – DBC is a public private partnership organization that need to make sure the Netherlands is leading in the field of blockchain worldwide.

Swarm City - Advisory Board Member – SC is one of the earlier and larger Ethereum startups. Building a platform to support for the sharing economy

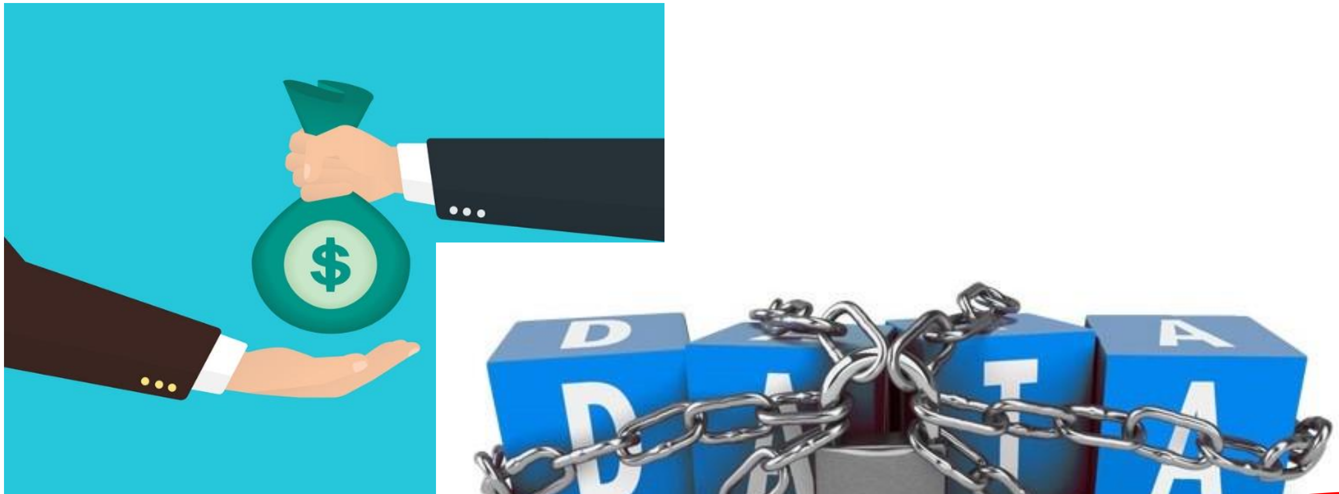
ISO – Smart Contract Standardisation group member – spokesperson for NL

Techruption Blockchain Incubator – blockchain & business expert – helping various startups as coach/reviewer

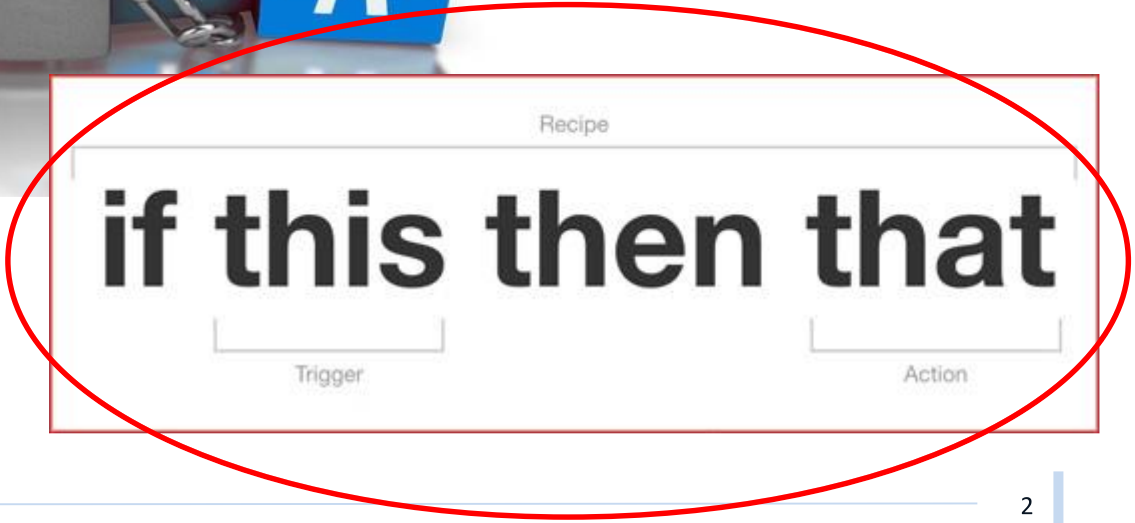
OurSurance – Founder, a blockchain based peer 2 peer insurance company, based on Ethereum blockchain/smart contracts

What is the potential use of blockchain

Blockchain can be used to virtually change everything to peer-to-peer. In high level you can use is for:



Smart Contracts



Three main misconceptions on smart contracts

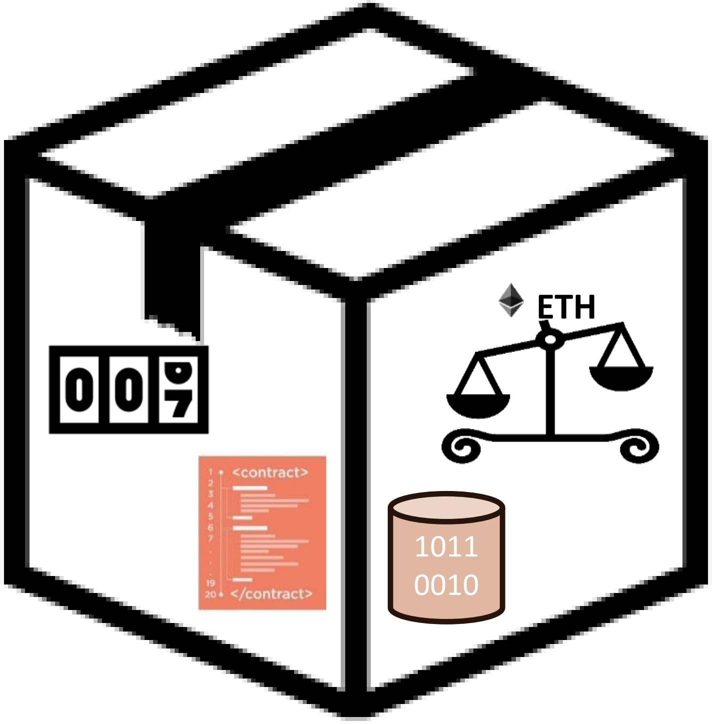
It’s just code not a contract – smart contracts can respresent the so called “operational semantics”. Especially as multiple parties activily transact to it, which can be seen as signing.

replaceAll(",", " ", a); a = a.
return a.split(" "); } $("#unique").
a = array_from_string($("#fin").
").val(), c = use_unique(array_from_
").val()); if (c < 2 * b - 1)
this.trigger("click");
!= a[b] && " " != a[b]
logged").val(); c = array_from_
c.length;b++) { -1 != a.indexOf(
for (b = 0;b < c.length;b++)
$("#User_logged").val(
click(function(

Smart contracts can work fully autonomously – smart contracts are transaction driven (thus reactive!). Cannot look outside their blockchain and even limited within their blockchain.

THE smart contract – There is no such thing as THE smart contract as there is no THE blockchain. Various blockchains have various smart contract capabilities. Look in your design at required elements.

Elements of smart contracts



A smart contract (or better, contract account) can be viewed as a “box” with the following elements:

- The nonce, a counter used to make sure each transaction can only be processed once
- The account's current ether balance
- ***The account's contract code,***
- The account's storage (empty by default)*

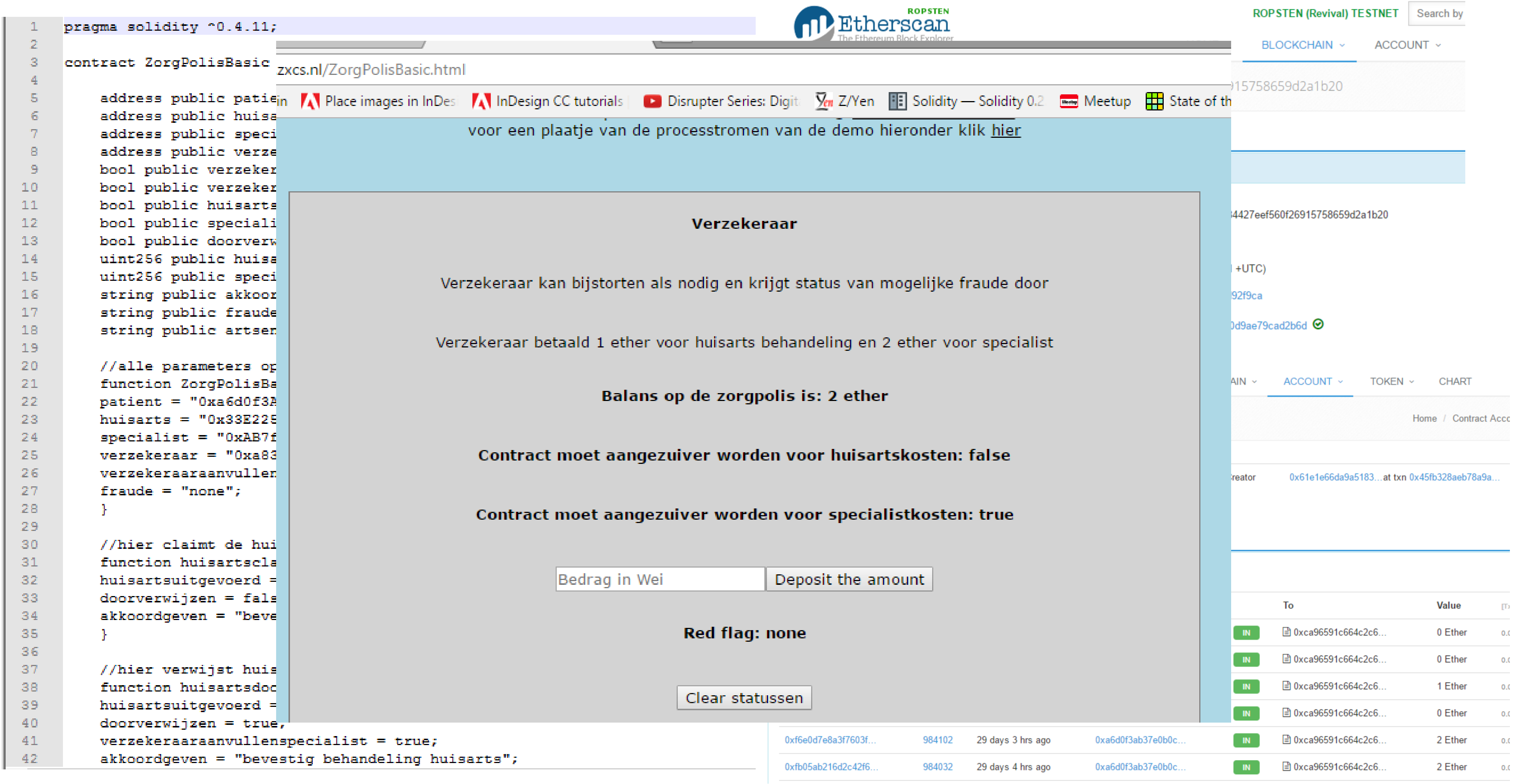
Every account has a 20-byte address e.g.:

0xa8323F5fBcf1980B2093a633cF03020900B81d53

State transitions are direct transfers of value and/or information between accounts.

*Data is stored in 3 spaces within Ethereum: Stack, Memory and Storage. The first two are reset after computation.

Example smart contract



```
pragma solidity ^0.4.11;

contract ZorgPolisBasic

    address public patie
    address public huisa
    address public speci
    address public verze
    bool public verzeker
    bool public verzeker
    bool public huisarts
    bool public speciali
    bool public doorverw
    uint256 public huisa
    uint256 public speci
    string public akkoor
    string public fraude
    string public artsen

    //alle parameters op
    function ZorgPolisBa
    patient = "0xa6d0f3A
    huisarts = "0x33E225
    specialist = "0xAB7f
    verzekeraar = "0xa83
    verzekeraaraanvullen
    fraude = "none";
    }

    //hier claimt de hui
    function huisartscla
    huisartsuitgevoerd =
    doorverwijzen = fals
    akkoordgeven = "beve
    }

    //hier verwijst huis
    function huisartsdoc
    huisartsuitgevoerd =
    doorverwijzen = true;
    verzekeraaraanvullenspecialist = true;
    akkoordgeven = "bevestig behandeling huisarts";
```

zxcs.nl/ZorgPolisBasic.html

voor een plaatje van de processtromen van de demo hieronder klik hier

Verzekeraar

Verzekeraar kan bijstorten als nodig en krijgt status van mogelijke fraude door

Verzekeraar betaald 1 ether voor huisarts behandeling en 2 ether voor specialist

Balans op de zorgpolis is: 2 ether

Contract moet aangezuiver worden voor huisartskosten: false

Contract moet aangezuiver worden voor specialistkosten: true

Red flag: none

915758659d2a1b20

34427eef560f26915758659d2a1b20

+UTC)

92f9ca

0d9ae79cad2b6d ✓

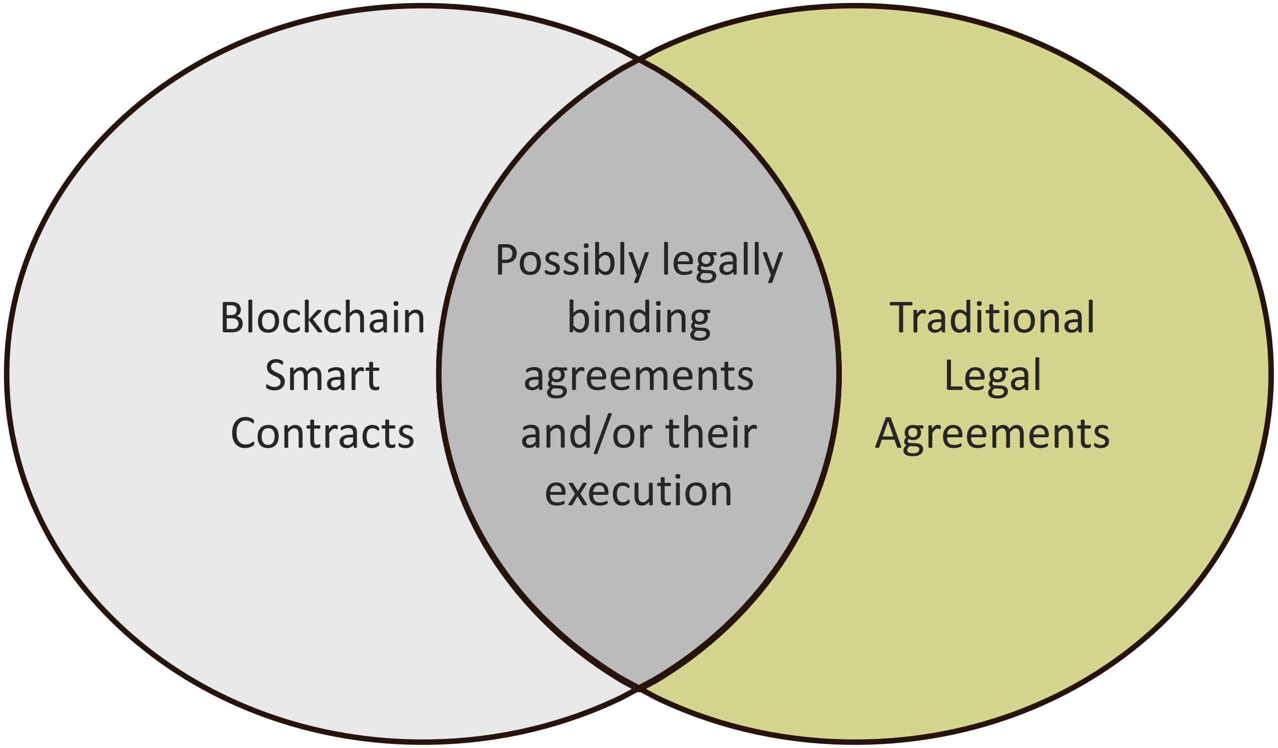
Creator 0x61e1e66da9a5183... at txn 0x45fb328aeb78a9a...

					To	Value
				IN	0xca96591c664c2c6...	0 Ether
				IN	0xca96591c664c2c6...	0 Ether
				IN	0xca96591c664c2c6...	1 Ether
				IN	0xca96591c664c2c6...	0 Ether
0xf6e0d7e8a3f7603f...	984102	29 days 3 hrs ago	0xa6d0f3ab37e0b0c...	IN	0xca96591c664c2c6...	2 Ether
0xfb05ab216d2c42f6...	984032	29 days 4 hrs ago	0xa6d0f3ab37e0b0c...	IN	0xca96591c664c2c6...	2 Ether

Smart Contracts vs Legal Contracts

a deterministic computer program deployed on a blockchain.

can have legal meaning, but not necessarily.


Blockchain Smart Contracts

Possibly legally binding agreements and/or their execution

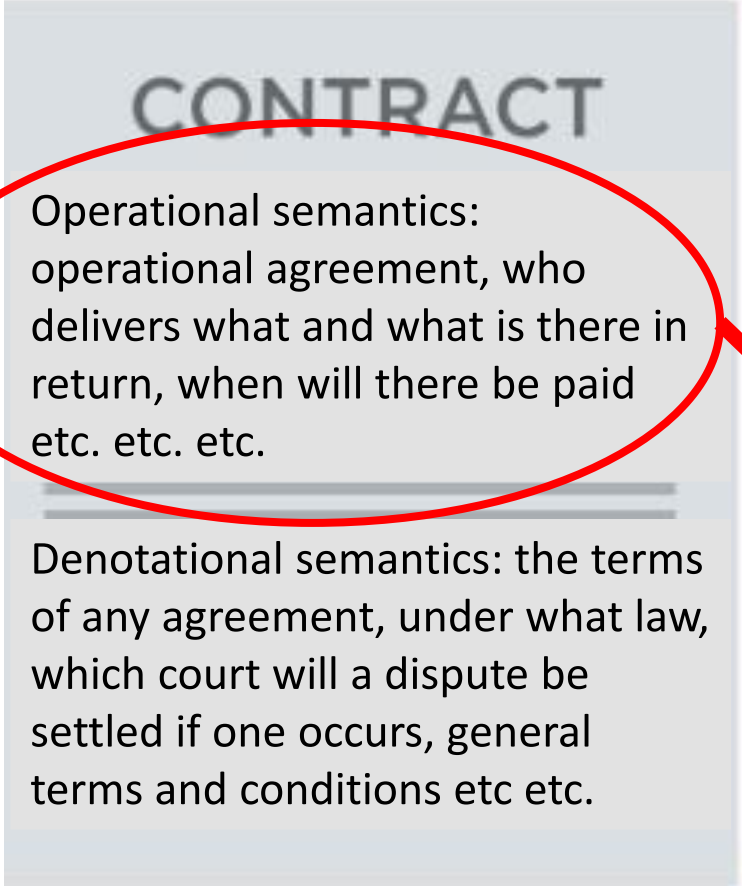
Traditional Legal Agreements

where transformation is suitable, recommended only for execution.

not all judgement is suitable for transformation into coding.

Operational vs Denotational Semantics and Smart Contracts

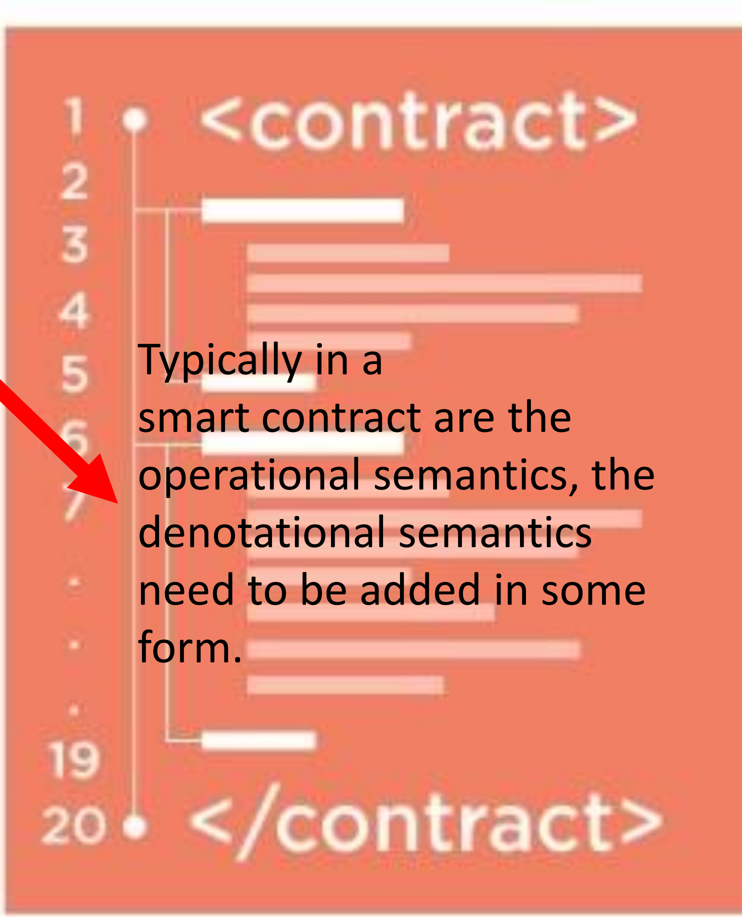
Possibly legal binding smart contracts



Operational semantics: operational agreement, who delivers what and what is there in return, when will there be paid etc. etc. etc.

Denotational semantics: the terms of any agreement, under what law, which court will a dispute be settled if one occurs, general terms and conditions etc etc.

Traditional contract



Typically in a smart contract are the operational semantics, the denotational semantics need to be added in some form.

Smart contract

Legal Pointers for smart contracts

If one specifically wants to create a whole agreement in code, the purpose of the code should at least be written in formal language as well and be distributed to parties involved,

At all times in legally binding smart contracts, upfront, the factual and legal possibilities to
a) dispute resolution and
b) the terms for automatic execution (e.g. authorisation of both parties) should be thought through.
c) one should upfront also consider things like jurisdiction etc.

The difference between smart contracts on a permissioned or permissionless blockchain is important as these can have completely different governance models and accountability issues.

PERMISSION GRANTED

Smart? Trusted sources of information!

Smart contracts are always triggered by a message or transaction. One should always be aware where the initiation for the message comes from as once triggered, the exectution is irreversable!

One can use “Oracles” that could trigger smart contracts. In the Netherlands, we have various usable oracles (e.g. KNMI database, BRP, etc).

Also IoT is seen as a potential very usefull way to trigger smart contracts, but one should always be aware that sensoring can create false signals as well (and thus false triggers for a smart contract).

An alternative can still be an “Oracle” by voting of the individuals involved in the contract or human “Oracle”.

THE smart contract?

Smart contracts vary per blockchain!

Bitcoin

Not designed for smart contracts! (but possible e.g. via sidechains)

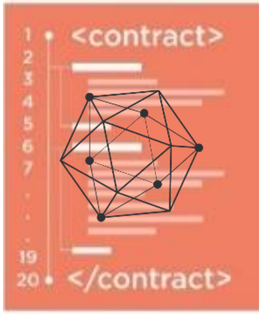
≠

Ethereum

Full smart contracts, tokens, crypto. Mainly solidity.

≠

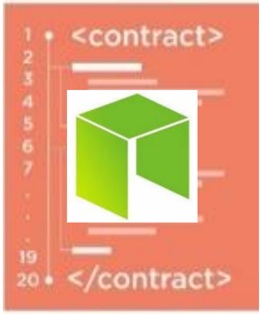
Hyperledger



Smart contracts, NO cryptocurrency

≠

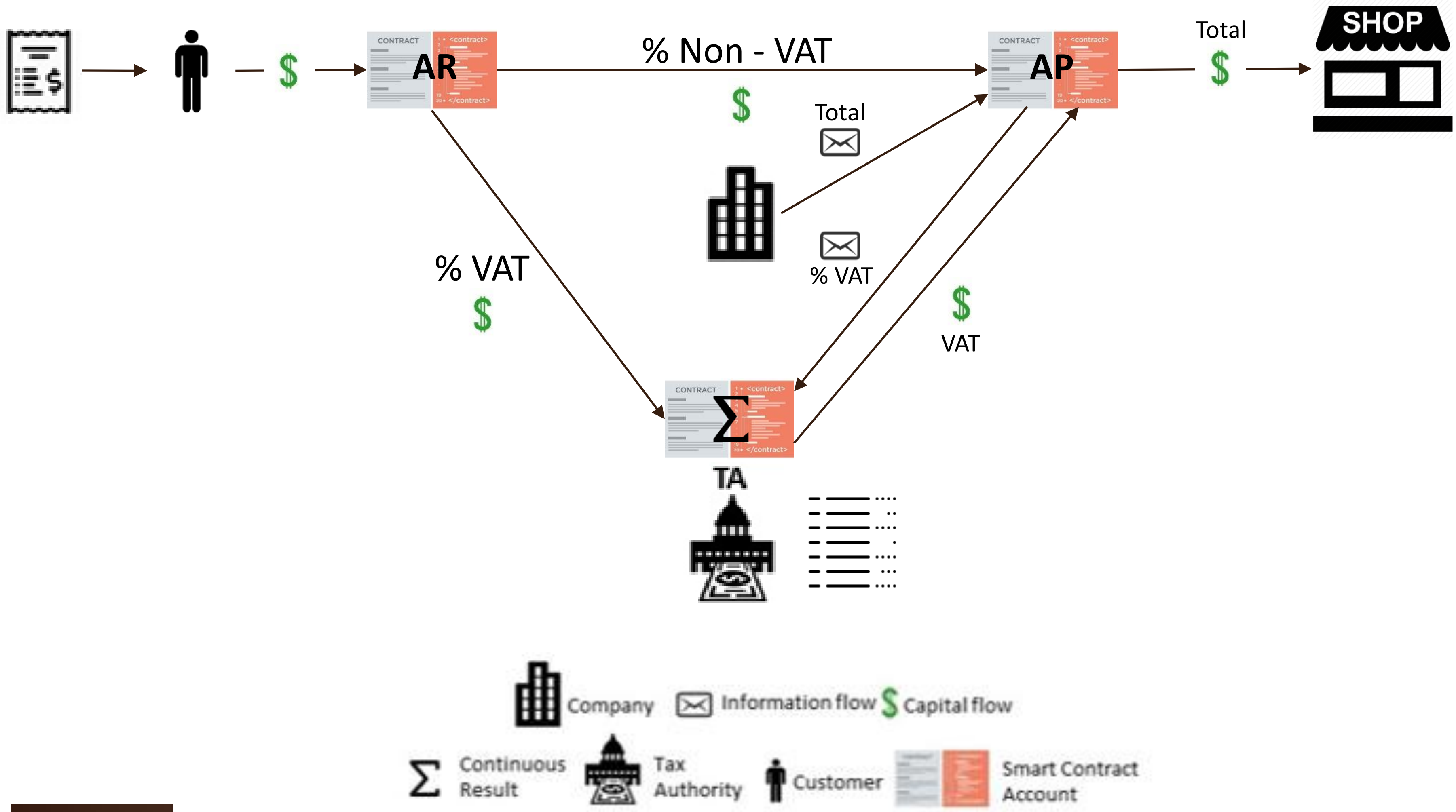
NEO



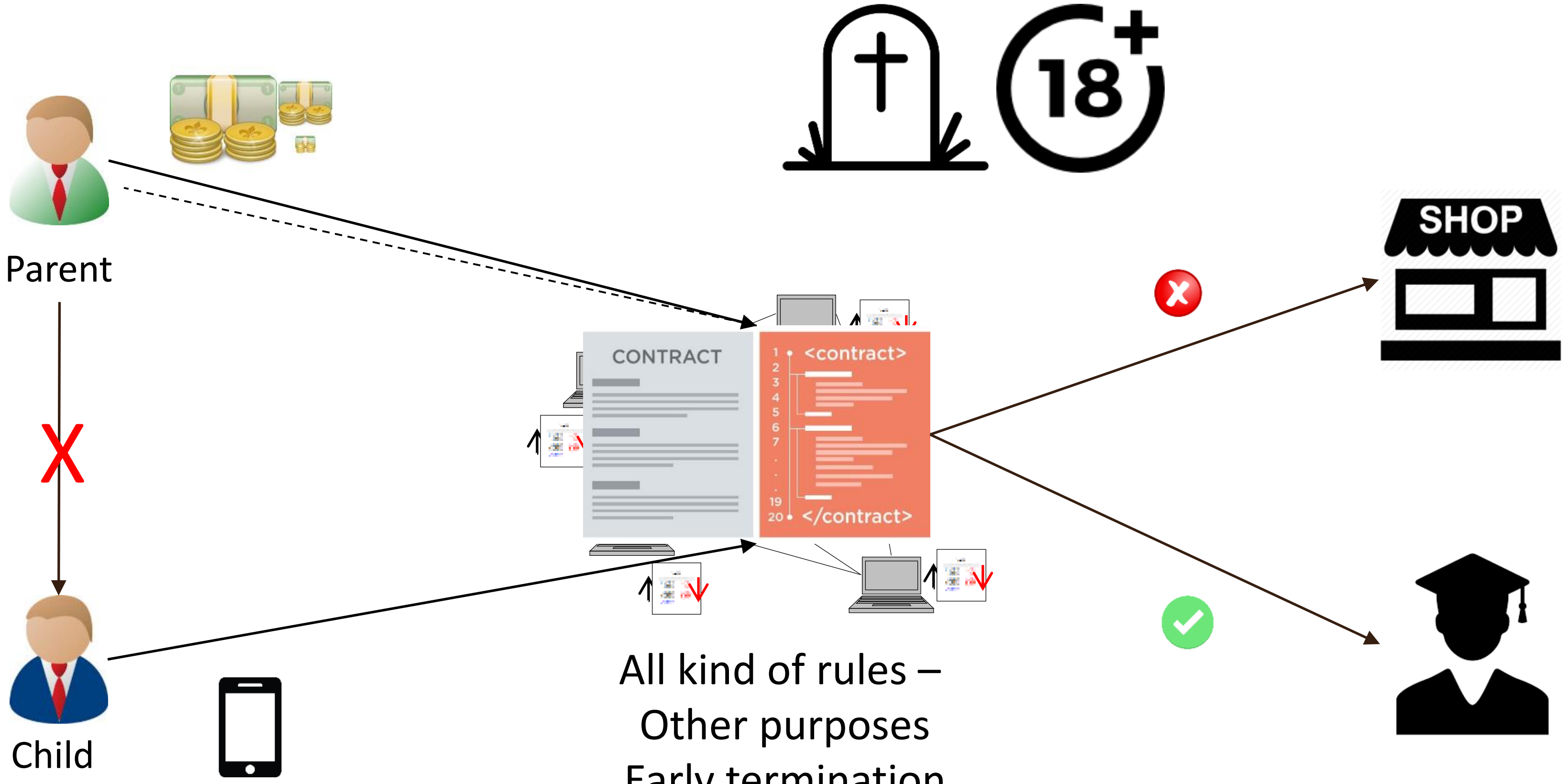
Full smart contracts, tokens, crypto. Various languages. Early stage!

Potential effect on business models

Blockchain/Smart Contract Real Time VAT



Allowance smart contract example

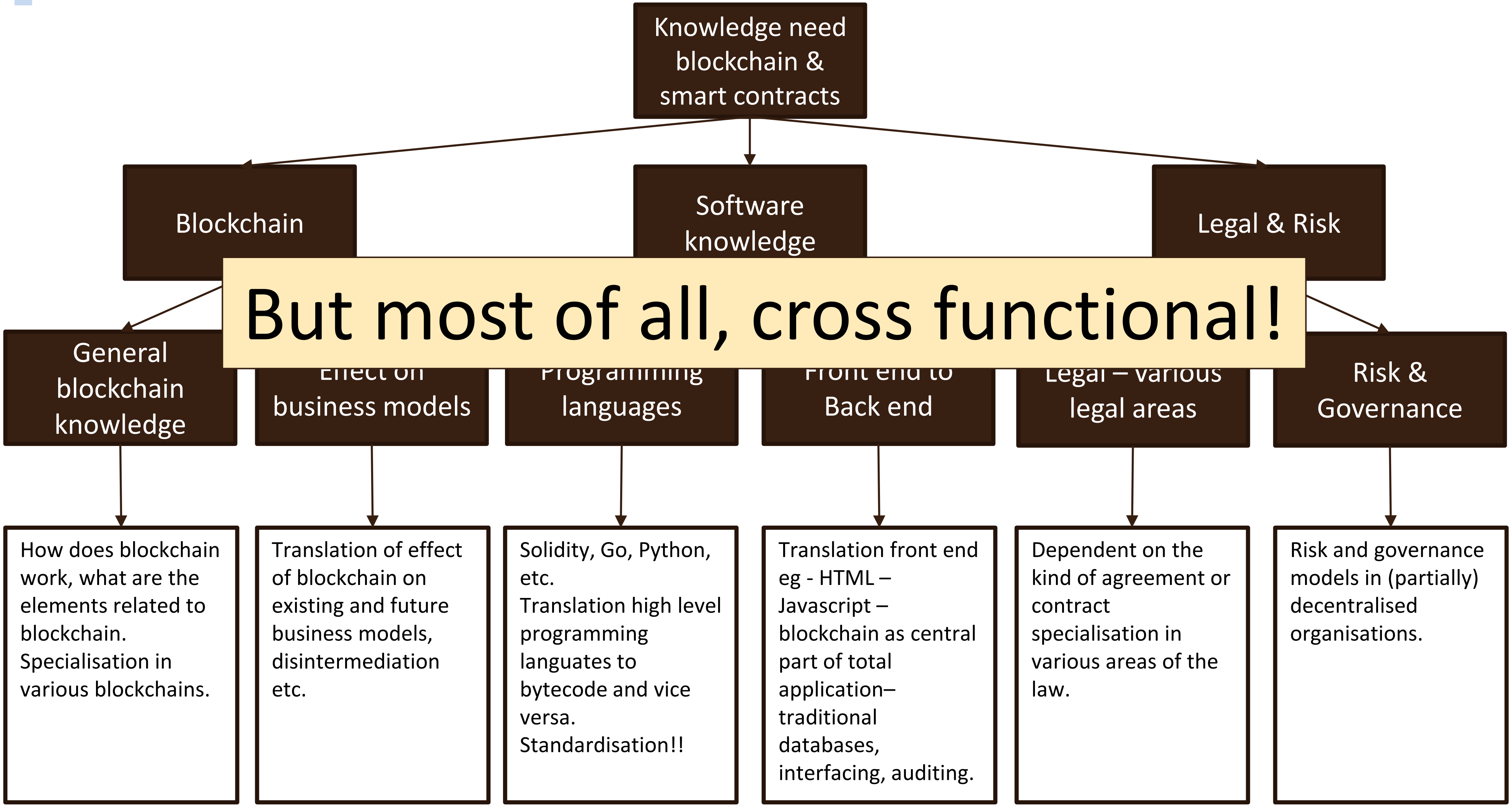


All kind of rules –
Other purposes
Early termination
Etc.

Towards robust smart contracts and interesting developments

What do we need to create robust and sustainable smart contracts

Observations: future knowledge requirements



Knowledge need blockchain & smart contracts

Blockchain

Software knowledge

Legal & Risk

But most of all, cross functional!

General blockchain knowledge

Effect on business models

Programming languages

Front end to Back end

Legal – various legal areas

Risk & Governance

How does blockchain work, what are the elements related to blockchain. Specialisation in various blockchains.

Translation of effect of blockchain on existing and future business models, disintermediation etc.

Solidity, Go, Python, etc.
Translation high level programming languates to bytecode and vice versa.
Standardisation!!

Translation front end eg - HTML – Javascript – blockchain as central part of total application– traditional databases, interfacing, auditing.

Dependent on the kind of agreement or contract specialisation in various areas of the law.

Risk and governance models in (partially) decentralised organisations.

Building Robust and Secure Smart Contracts - Elements of good smart contracts

Should contain:
Clear terms on execution and transfer of payment (when do we pay, what are the payment terms)
If applicable legal considerations / context – either in the comments of a smart contract or hashed as a state variable with a reference to where the original can be found.

MUST HAVE

Compliance
Risk
Governance

Should have considered:

Compliance considerations
Risk and Fraud considerations
Privacy considerations
Unestablished legality
Dispute resolution

Building Robust and Secure Smart Contracts - Governance and upgrading smart contracts

Make sure that you build in proper access control (through modifiers) and if needed multisig constructions (e.g. multiple ok's from different persons in order to update a variable).

Although the code can't be altered once deployed, variables can be updated. Design your contracts in such a way that you have flexibility through variable updates.

"Proxy function", keep an open design function that can point to a different smart contract which can be added later to add functions through an additional smart contract.

Don't rule out the possibility for human interference/escape hatch, e.g. in the case of dispute resolution that a third party can make a final ruling in case of dispute.

Building Robust and Secure Smart Contracts – some best practices

Open Source World – so find out what has been built before. MIT-Labs etc.

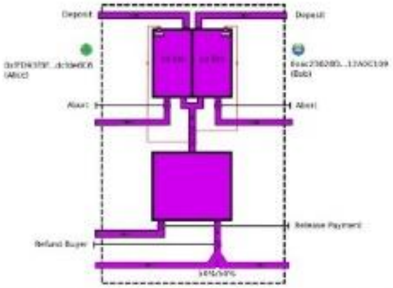
Keep smart contracts small and simple – reduced attack surface and easier to reason and scan. (Also cheaper!)

Traditional software development best practices like defensive programming, fuzzing and automated tests and frameworks and code reviews and audits are still very usefull and valuable!

Partially from "Will that smart contract really do what you expect it to do"by Everts and Muller

Smart Contract enabling blockchains

And some interesting developments



Zero-knowledge proof
A
B

Questions?

Smart contracts

as a specific application of blockchain technology

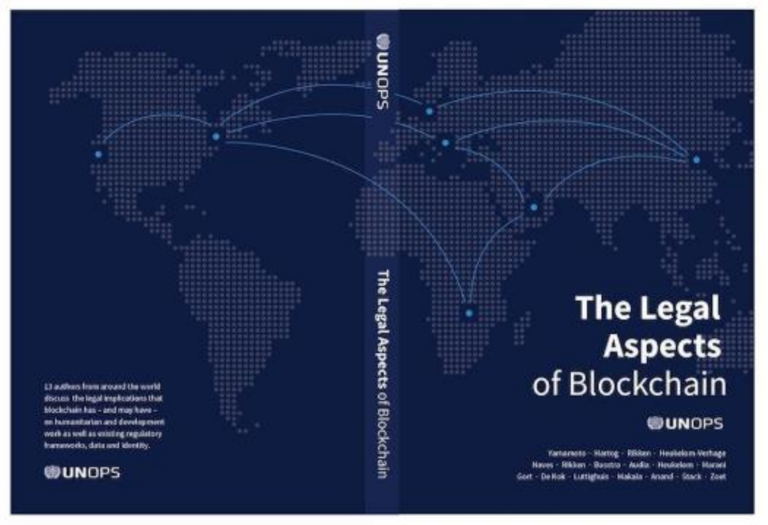
First reconnaissance of questions relating to legislation, regulations and future knowledge needs as a consequence of blockchain technology and more specifically, smart contracts.

Dutch Blockchain Coalition

connect and create

Smart Contract Working Group – Dutch Blockchain Coalition

www.dutchblockchaincoalition.org



13 authors from around the world discuss the legal implications that blockchain has – and may have – on humanitarian and development work as well as existing regulatory frameworks, data and identity.

UNOPS

UNOPS

The Legal Aspects of Blockchain

The Legal Aspects of Blockchain

UNOPS

Yamamoto · Hartog · Rikken · Heukelom-Verhage

Naves · Rikken · Bosstra · Audia · Heukelom · Marani

Gort · De Kok · Luttighuis · Makala · Anand · Stack · Zoet

Olivier Rikken

Director Blockchain & Smart Contracts

+31 611 394 292

orikken@axveco.com | www.axveco.com