Smart Contracts and the need for Semantics (and Reasoning)

Rieks Joost, Maarten Everts and Christopher Brewster

IofA Conference, London, 16 Oct 2017.

Outline

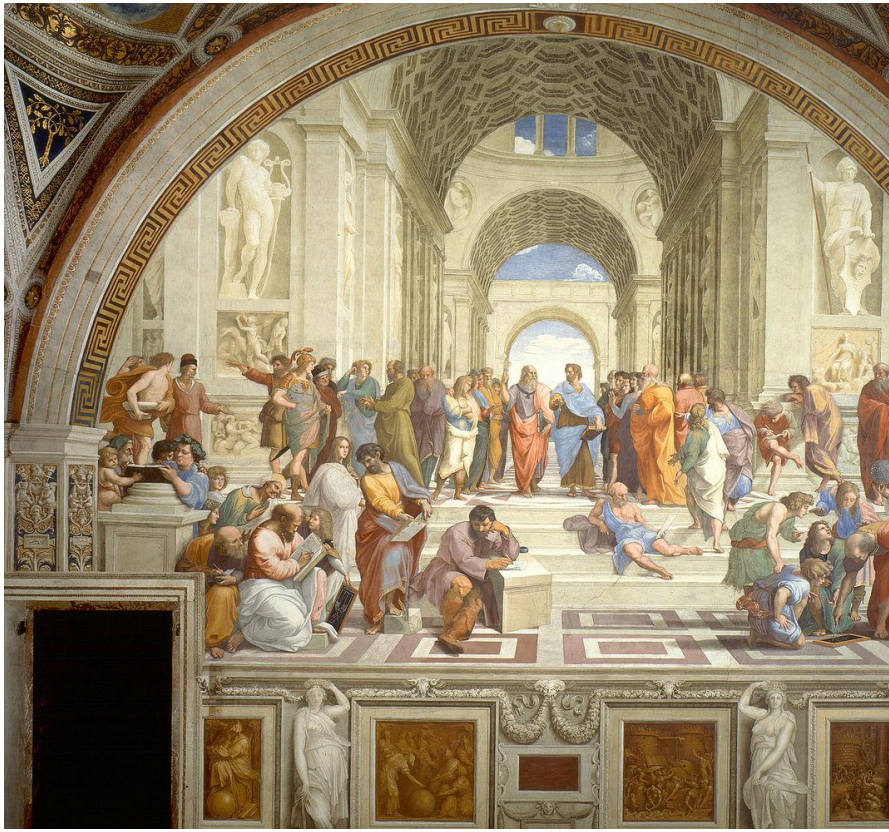
- Types of Contracts
- Layers of semantics
- Electronic Business Transactions
- Meaning of statements (in contracts)
- Building blocks
- Ontologies of legal concepts
- Standards and ontologies for domains
- What next?

A simple typology of contracts

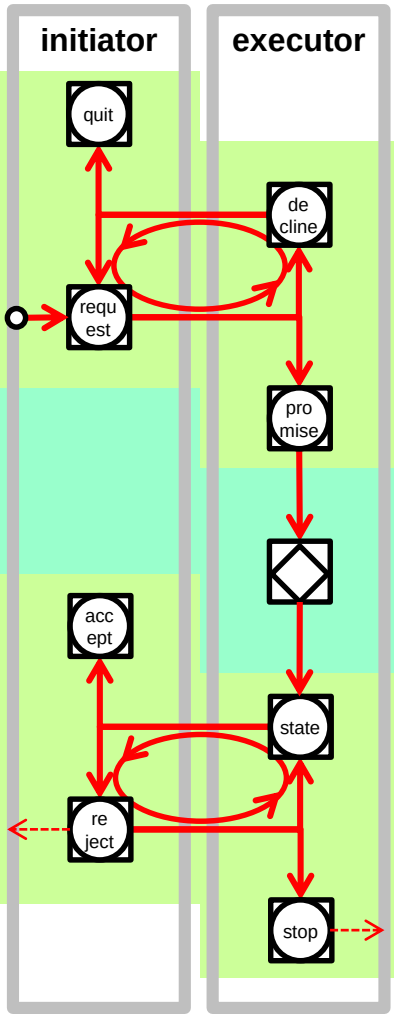
1. Face to face contracts – common understanding of trust with friends, family and close colleagues
2. Face to face + written contract (e.g. a traditional tenancy agreement – recourse to law too costly)
3. Face to face + written contract + legal framework/arbitration framework (e.g. **normal business practice**)
4. Written Digital contract/written contract + legal framework/arbitration framework (e.g. ebay purchases)
5. **Ricardian “smart contracts” = "smart contract" code (logical contract + written contract) + arbitration framework**

Layers of semantics

- Multiple layers in building an Internet of Agreements infrastructure
- Each layer needs semantics clearly defined, to allow reasoning
- Layers include (at least):
  - The Blockchain protocol layer
  - The “smart contract code” layer (software language used for contracts)
  - The terminology layer (terms and identifiers)



Electronic Business TRANSACTIONS



- **Phase 1 – proposition phase**: Initiator and executor negotiate the transaction agreement, and **decide** to either quit, or commit.
- **Phase 2 – execution phase**: parties fulfil their obligations
- **Phase 3 – result phase**: Executor and initiator negotiate acceptance of the results, and **decide** to either accept, or escalate.

Terminology Definition process

- Design & Engineering Methodology for Organizations
  DEMO (Dietz, J., TU-Delft);

- TNO's **Terminology specification** method
  for constructing and maintaining '**definitions**' (terminology)
  that demonstrably mean the same thing
  for all parties that are involved
  in a particular context, so that
  the can precisely define
  their **semantics**.

- **semantic web technology**
  e.g. RDF(S), JSON-LD, etc.

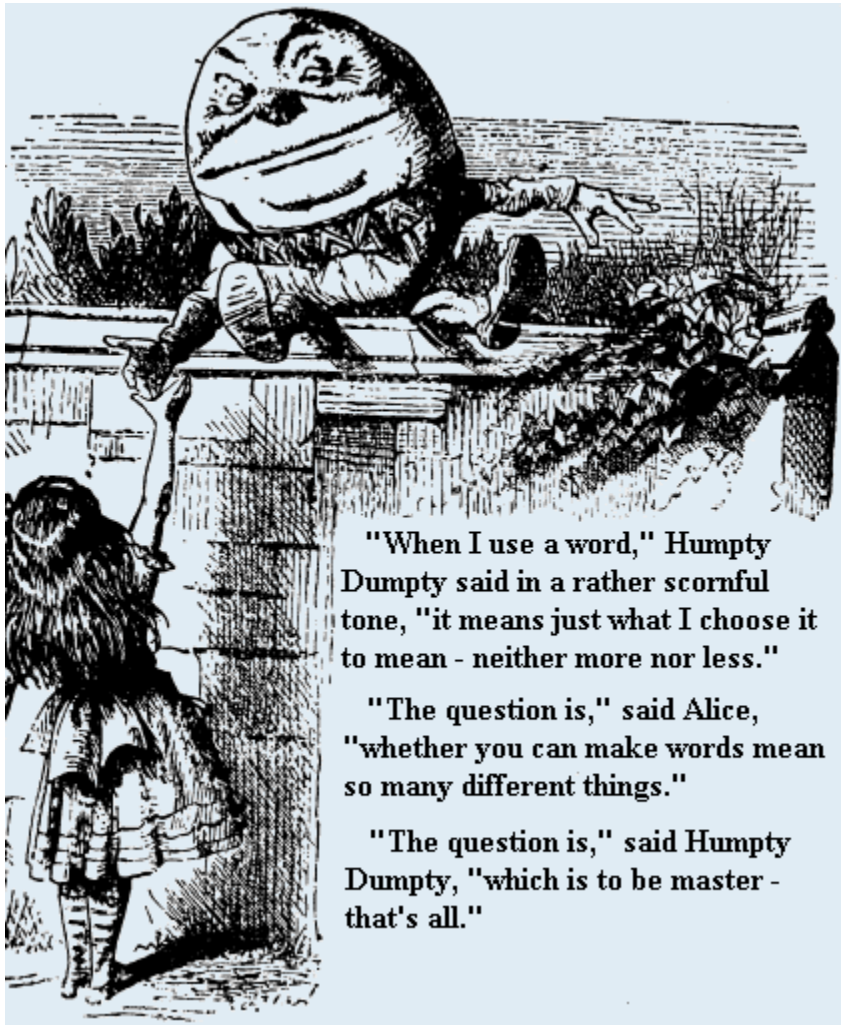
Electronic **BUSINESS** Transactions

- A business will generally commit to a transaction when
  - the **value** of what it gets outweighs the value of what it invests;
  - the **risk** of engaging in the transaction is acceptable;
  - the position you have in case of a **dispute**, is sufficiently good.

- Committing to a transaction is a business decision that requires
  - **data** (statements, e.g. about the customer);
  - **business logic** (that processes this data to reach a conclusion);
  - data and business logic to be **valid**.

Invalid business decisions increase business risk

Meaning and truth (of Statements)



Caroll, L.: Alice's Adventures in Wonderland, 1865

- The **meaning of a statement** is subjective, thus requiring the business to decide this:
  - if not, there is incoherence;
  - generally accepted meanings can be used;
  - Ontologies if specified **make life easier**.
- The **truth of a statement** is also subjective, thus requiring the business to decide this, e.g.:
  - after (proper) investigation;
  - by relying on what others say (that are trusted to state this).

The Uses of Reasoning 1

- **PRE-EXECUTION**
- Reasoning and formal proofs can **help** to ensure a piece of smart contract code corresponds to **intent. Guarantees are very difficult.**
  - Can help to show there are no “loopholes” in a smart contract code.
  - Can help to show that separate (locally scoped) rules do not interact to create global contradictions, or be combined in unforeseen ways and have unwanted side effects.
- Formal methods are not a silver bullet – a tool
- Formal specifications will also allow for more confidence in combining “patterns” for smart contracts
- → **Formally rigorous design patterns for smart contracts are needed.**

The Uses of Reasoning 2

- **POST EXECUTION**
- Reasoning can be used to verify contract completion. Combination of input from “executor” of contract together with other data input to conclude YES contract has been fulfilled, or NO contact has not been fulfilled
- Context-aware systems (ubiquitous computing) will play an increasing role.
- Allow inferences like:
  - Executor has sent 100kg carrots
  - Carrots are of type “Imperator 58”
  - Carrots are now in location XYZ (street address, or long/lat)
  - Therefore contract has been fulfilled
- The more there are semantics, the more the logical rigour, the more this process can be automated.
- **And arbitration can be avoided.**

Building Blocks towards semantics in “smart” contracts

1. Efforts to provide logical rigour (formal verification) to blockchain platforms
   - Tezos (https://www.tezos.com/) for example – uses OCaml due to logical rigour and type logic
   - Tezos also provides a smart contract language (Michelson) that is amendable to formal verification.

2. Open source and commercial reasoning engines
3. Existing ontologies and related work for formalisation of legal contracts

3. Standards (i.e. vocabularies of varying degrees of formality) and ontologies for the electronic transactions

Semantics for legal concepts and contracts

- Existing ontologies and related work for formalisation of legal contracts (e.g. Casellas 2011, Casanovas et al. 2016 or for a specific example concerning tenders Distino 2016).
  - Pioneer here was Joost Breuker – Leibniz Centre for Law – no longer active.
- Most recent work has concerned ontologies used for Information Extraction over legal texts.
- Smart contracts (as Riccardian contracts) provide a HUGE OPPORTUNITY to explore more thoroughly use of semantics and reasoning for contract management.

Standards and Ontologies for commercial transactions

- Start with Schema.org (http://schema.org/) and GoodRelations Ontology (http://www.heppnetz.de/projects/goodrelations/) designed for e-commerce but widely used.
- Existing vocabularies for supply chain activities (e.g. GS1 GTIN for identifiers, GS1 EPCIS for processes (cf. Solanki and Brewster 2015 for a formalisation)
- Standards for geographical location management and inference (e.g. Geonames)
- Choose a specific domain to test out e.g. pharmaceuticals, logistics, agrifood etc.
  - Many standards exists, low uptake in practice.
  - Scott Nelson argues blockchains ecosystem an opportunity for incentivising adoption ….

What next?

- TNO has a great deal of expertise in **TERMINOLOGY DEFINITION PROCESSES** i.e. getting a group/community to agree a terminology for a domain, formalise this and manage the ontology/standard.
  - We have standards management tools backed by relation algebra.
- TNO experience with relation algebra (RA) for rigorously defining (closed world) ontologies – provides mathematical rigor
- TNO also has experience in formal verification for smart contract code.
- We also have developed standards for Internet of Things (ETSI standard), the Dutch national temporary staffing standards, Dutch national invoicing standards.

Questions

Christopher.Brewster@tno.nl

Acknowledgements

This research is support by the Techruption Blockchain project (https://blockchain.tno.nl/projects/techruption/)
For further information consult http://blockchain.tno.nl

References

- Casellas, N. (2011). *Legal Ontology Engineering: Methodologies, Modelling Trends, and the Ontology of Professional Judicial Knowledge*. Springer Science & Business Media
- Casanovas, P., Palmirani, M., Peroni, S., van Engers, T., & Vitali, F. (2016). Semantic web for the legal domain: the next step. *Semantic Web*, *7*(3), 213–227. Retrieved from http://content.iospress.com/articles/semantic-web/sw224
- Distinto, I., d'Aquin, M., & Motta, E. (2016). LOTED2: An ontology of European public procurement notices. *Semantic Web*, *7*(3), 267–293. Retrieved from http://content.iospress.com/articles/semantic-web/sw151
- Solanki, M., & Brewster, C. (2015). Linked Pedigrees - Enabling real time data visibility in agri-food supply chains. *International Journal on Semantic Web and Information Systems*, *10*(3).