A lightweight BPMN Execution Engine on Ethereum

Luciano García-Bañuelos
University of Tartu
Blockchain ...
From the point of view of this talk

- Distributed, highly (fully?) replicated, append only database
- Each record that is inserted into the database is secured from tampering and revision
- **Keywords**
  - **Distributed**: The database is highly replicated, following a peer-to-peer architecture (there is not a central master node)
  - **Append-only storage**: Writing to the database can only happen by adding new records at the end of it
  - **Descentralized consensus**: the majority of nodes needs to agree that a yet-to-be-added record is legitimate or not
Assume “Alice” sends some bitcoins to “Bob”, a bitcoin transaction will accordingly store:

- An input: This is a record of which bitcoin address was used to send the bitcoins to “Alice” in the first place
- An amount: The amount of bitcoins being traded
- An output: This is “Bob”’s bitcoin address
Ethereum’s take on blockchain

Ethereum components:
1) A distributed blockchain (similar to Bitcoin)
2) A computing platform that enables the execution of Smart Contracts
   The above is done on top of the Ethereum Virtual Machine (EVM)

Smart contract ...

• Programs living on the blockchain (e.g. Ethereum) with their own memory and code

• Blockchain transactions via execution of contract functions (clauses)

• Manipulate persistent data (contract state), transfer ether, interact with other contracts or with “agents”
Blockchain as a platform for collaborative business processes

- Participants agree on a collaborative process and a model for it
- The model is translated to a smart contract(s) to be executed on the blockchain
- Smart contracts listen to process execution events and interact with agents or other smart contracts in order to monitor and/or execute the process
Blockchain as a platform for collaborative business processes
Blockchain as a platform for collaborative business processes

- The model is translated to a smart contract(s) to be executed on the blockchain
Blockchain as a platform for collaborative business processes

- Smart contracts listen to process execution events and interact with agents or other smart contracts in order to monitor and/or execute the process.

![Diagram of blockchain platform for collaborative business processes](image_url)

3.2 Design Time: Translator

The translator is used at design time: it takes an existing business process specification as input and generates smart contracts. These implement the C-Monitor or mediator and can be deployed and executed on the blockchain. In a collaborative process, this functionality must be split and distributed between the smart contract and the triggers. The translator creates the artifacts in such a way that the triggers and the smart contract can collaborate directly with each other over the blockchain network.

When the translator is called, it may not be known which participants will play which roles. Therefore, the translator outputs only a factory contract, which in turn contains all information needed for instantiating the process. The factory contract includes the methods for instantiation and two types of artifacts: (i) an interface specification per role (e.g., buyer, manufacturer, and shipper) in a collaborative process, to be distributed to the respective triggers, and (ii) a process instance contract, which is deployed to the blockchain when the process is instantiated. The process instance contract contains the implementation of the business logic and takes the form of a C-Monitor or mediator, depending on the content of the original process specification.

The overall translation algorithm has two phases. First, the translator parses the input process model and iterates through all its elements, where it generates two lists per element in the process model: one list of previous elements and one of next elements. Then, the translator translates each element with its respective links, generating Solidity code based on the translation rules for different types of elements as detailed in the TR.

Note that, in the current implementation, only some combinations of consecutive gateways can be connected to each other without tasks in between. The previous element list is used by the translator to determine which other elements need to be deactivated when the current element...
Smart contract ...

• Described by Nick Szabo in the late 90s

The basic idea behind smart contracts is that many kind of contractual clauses (e.g. collaterals, bonding, delineation of property rights) can be embedded in hardware and software in such a way as to make breach of contract expensive for the breacher
Smart contract?

• Contract: Traditional way to formalize a relationship

• Definition from Merriam-Webster
  ◦ A binding agreement between two or more persons or parties, especially one legally enforceable
  ◦ A business arrangement for the supply of goods or services at a fixed price
  ◦ The act of marriage or an agreement to marry

  ◦ A document describing the terms of a contract
A concrete example ...

- Within a limited amount of potential loss, the vending machine takes in coins and dispenses change and product according to the displayed price.

- Vending machine implements a sort of contract with bearer:
  - Anybody with coins can participate
  - Some security mechanisms protect stored coins and product from attackers
Another example: Remote purchase
Escrow (Smart contract)

Purchase contract created

Notify Buyer about contract creation

Confirm purchase (payment)

Notify Seller about payment

Confirm reception

Notify Seller about reception

Buyer

Seller

Purchase details

Create purchase contract

Confirm purchase (payment)

Ship product

Deliver product

Confirm reception

Buyer

Shipping company

Seller

Seller

Shipment details

Payment details

Shipping company

Buyer

Delivery details
Another example: Remote purchase
Remote purchase (cont)
Remote purchase ... middleman
Smart contract on BPMN
Demo time
Temporal constraints on BPMN

- Block host & guest deposit
- Cancellation by guest
- Cancellation by host
- 7 days before check-in