UNIVERSITÄT BERN

# SMART CONTRACTS – LEGAL IMPLICATIONS

PROF. DR. MIRJAM EGGEN

```
contract token {
    mapping (address => uint) public coinBalanceOf;
    event CoinTransfer(address sender, address receiver, uint amount);

  /* Initializes contract with initial supply tokens to the creator of the contract */
  function token(uint supply) {
        if (supply == 0) supply = 10000;
        coinBalanceOf[msg.sender] = supply;
    }

  /* Very simple trade function */
    function sendCoin(address receiver, uint amount) returns(bool sufficient) {
        if (coinBalanceOf[msg.sender] < amount) return false;
        coinBalanceOf[msg.sender] -= amount;
        coinBalanceOf[receiver] += amount;
        CoinTransfer(msg.sender, receiver, amount);
        return true;
```

*Source: bits on blocks*

# AGENDA

1. DEFINITION
2. EXAMPLE
3. OBLIGATION
4. TRANSFER
5. DEFAULT

# DEFINITION

IN THE CONTEXT OF BLOCKCHAINS, A SMART CONTRACT IS:
– PRE-WRITTEN LOGIC
– STORED AND REPLICATED ON A DISTRIBUTED PLATFORM
– EXECUTED BY A NETWORK OF COMPUTERS

*Source: bits on blocks*

## PROTOCOL

- Platform contract between platform operator and parties of SC
- No contractual relationship between these parties with open-source protocol

## APPLICATION

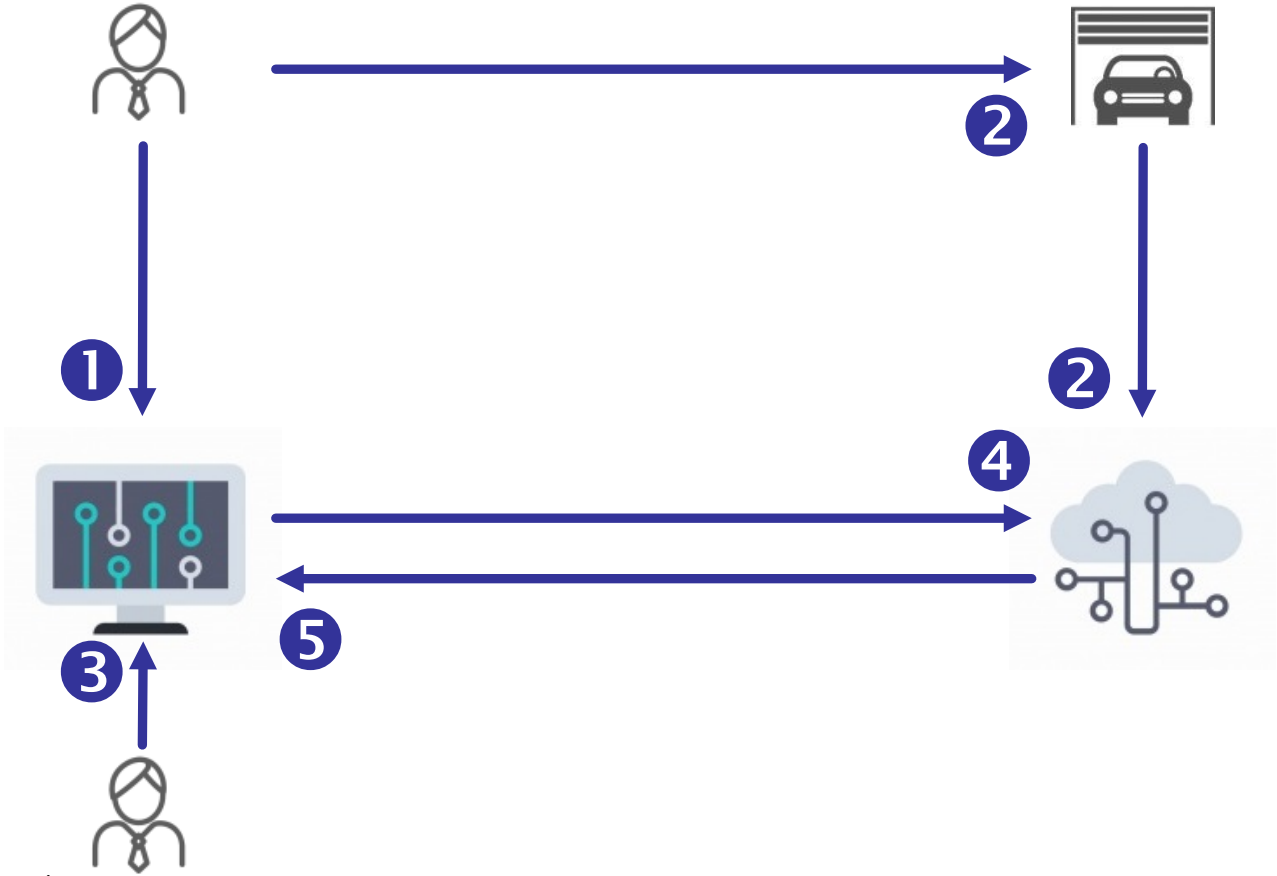
- Application contract between author of application and parties of SC

## SMART CONTRACT

- Applies to relationship between two parties e.g. buying goods

*Source: Furrer, Embedding of smart contracts*

# EXAMPLE




*Source: BlockchainHub*

# OBLIGATION

THE RELATIONSHIP BETWEEN A SC AND A CONVENTIONAL CONTRACT HAS TO BE ANALYZED CASE BY CASE. WHILE THE CONTRACT CAN BE DRAFTED IN CODE, MOST OFTEN THE PARTIES USE REFERRALS TO CREATE A LINK BETWEEN THE CONTRACT AND THE SC.

## CONSENSUS

- Only effective if specifically made part of the contract
- Accepting SC as a tool does not equal making the code part of the contract

## SOFTWARE

- In principle contractual provisions can be expressed as a computer code
- Parties must be able to gather information regarding the content of the contract

# TRANSFER

| | NATIVE | NON-NATIVE |
|---|---|---|
| FEATURES | ▪ Do not entitle outside of the blockchain<br>▪ Do not represent relative nor absolute rights | ▪ Entitle outside of the blockchain<br>▪ Represent relative or absolute rights |
| TRANSFER | ▪ Provide factual power | ▪ **Movable Property**: Smart Property, Besitzeskonstitut, BEG<br>▪ **Claim**: Novation, contract transfer, BEG<br>▪ **Security**: Besitzanweisung, BEG |

# DEFAULT

## ANTICIPATE AND CORRECT DEFAULT IN PERFORMANCE

CAREFULLY DESIGN AND TEST THE SMART CONTRACT

INTERFACE TO INDEPENDENT ARBITRATION BOARD

CONTRACTUAL DISTRIBUTION OF LIABILITY FOR FAULTS