

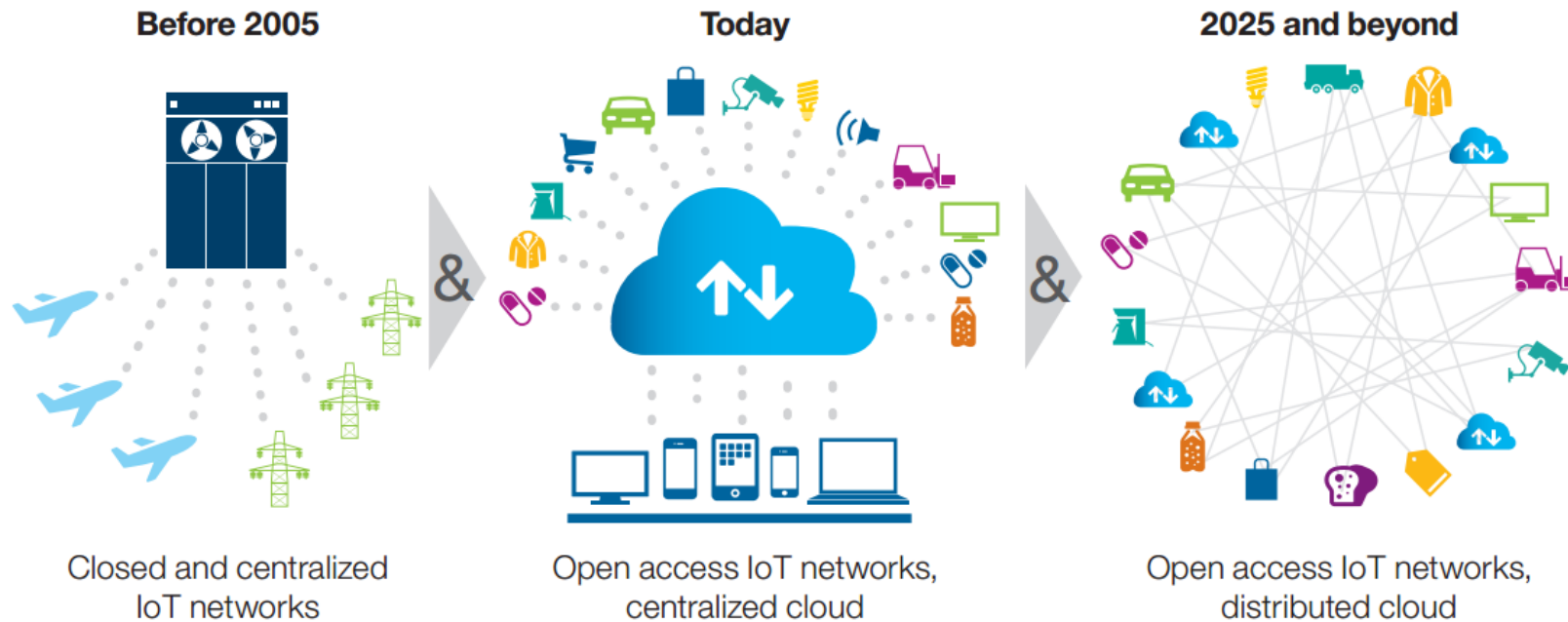
Business from technology

# Blockchain Technology and IoT New Challenges

IoT Week  
05 June 2018, Bilbao

Visa Vallivaara,  
Mathematician & Cyber Security Scientist  
VTT Technical Research Centre of Finland

# Blockchain & IoT



- Blockchain enables collaboration with very different kind of actors/machines.
- Smart machines will be more independent in the future and are able to take care of maintenance and bargaining by themselves

## Smart Contracts

- Smart Contracts are special transactions enabled by the second generation blockchains: Ethereum, Hyperlegder, Corda,...
- They are programmed to handle everything automatically under given conditions
- Pre-written logic, stored and replicated on a blockchain, self-executing by running the code, can enforce the code to make payments, update blockchain, transfer ownership, etc



# Possibilities of smart contracts in IoT

- Sharing of services and resources
- Automates time-consuming workflows in a cryptographically verifiable manner .
- Enables device autonomy, individual identity and integrity of data.



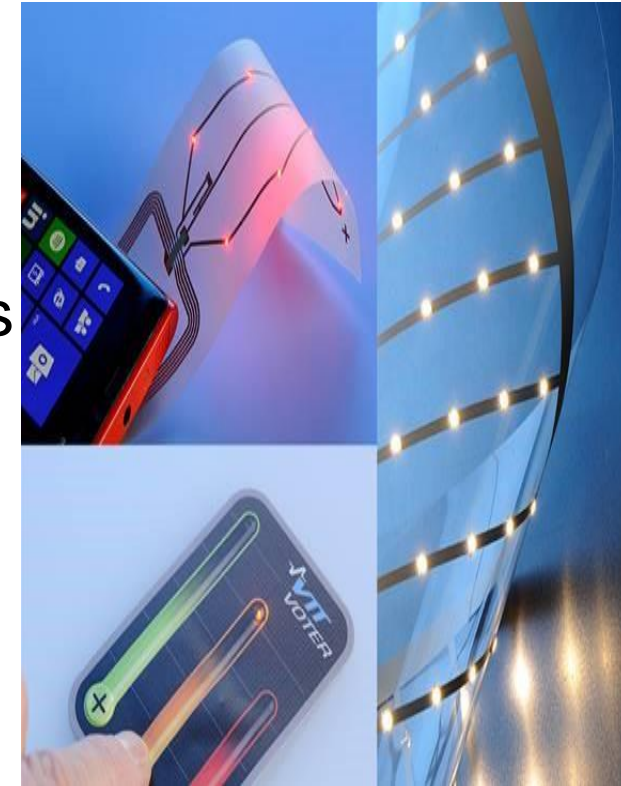
## Data Market

- Data is tradeable good
- Smart contracts allow complicated and automatic trade agreements without the need for trusted third party.
- Creators and users of data have different role.
- Combining digital objects with a smart contract would give the possibility to safely document all transactions.

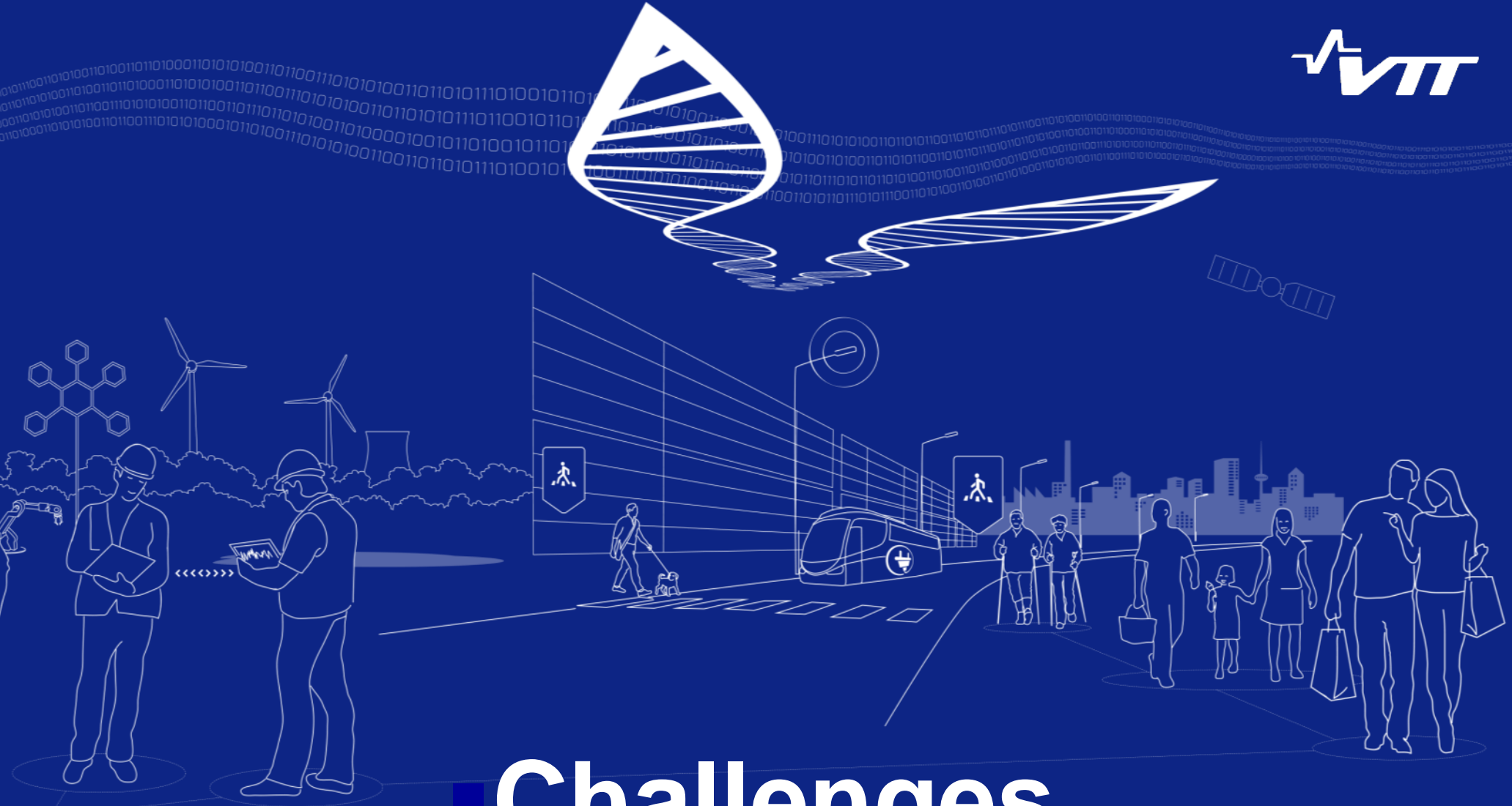


## Anti-Theft Sticker

- The tag is developed by VTT, Nokia and Streamr combined with a smart contract.
- The contract defines the terms and conditions of transportation and storage, and the fees.
- Small sensors are embedded in the tag for location, acceleration and temperature.
  - Constant monitoring of the terms in the smart contract
- Demo in the Consensus 2018, in New York.







# ■ Challenges

## Not very Smart Contracts

- They actually aren't smart or even contracts
  - In Hyperledger they are called Chain Code
- In the absence of a legal framework around Smart Contracts, it is uncertain who is liable for what if there is a failure of any sort.
  - “Code is Law”???
- Oracles — entities that provide this data — may not be trustworthy.
- The new middlemen will be the owners of the technologies that make Smart Contracts possible





## Scaling is a challenge



Kuva: <http://www.popularmechanics.com/culture/web/a11610/this-is-what-happens-when-a-bitcoin-mine-burns-down-17410755/>

# The DAO: Or How A Leaderless Ethereum Project Raised \$50 Million

Michael del Castillo (@DelRayMan) | Published on May 12, 2016 at 21:19 BST

FEATURE



862



611



20



626



0



DAO = Decentralized Autonomous Organization

Investments over 50\$ millions

*Ether* cryptocurrency


THE DAO IS CODE. |

No managers or board: "Code is Law"

Governed with smart contracts  
and voting of the shareholders

## Semantic DAO Code


```
function withdraw(uint amount) {  
    client = msg.sender;  
    if (balance[client] >= amount) {  
        if (client.call.sendMoney(amount)) {  
            balance[client] -= amount;  
        }  
    }  
}
```



Client wants to withdraw money


## Semantic DAO Code

```
function withdraw(uint amount) {  
    client = msg.sender;  
    if (balance[client] >= amount) {  
        if (client.call.sendMoney(amount)) {  
            balance[client] -= amount;  
        }  
    }  
}
```



Who?

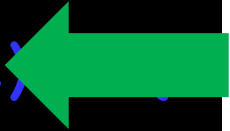
## Semantic DAO Code

```
function withdraw(uint amount) {  
    client = msg.sender;  
    if (balance[client] >= amount) {   
        if (client.call.sendMoney(amount)) {  
            balance[client] -= amount;  
        }  
    }  
}
```

Is there enough ethers in the account?

## Semantic DAO Code


```
function withdraw(uint amount) {  
    client = msg.sender;  
    if (balance[client] >= amount) {  
        if (client.call.sendMoney(amount))  
            balance[client] -= amount;  
    }  
}
```



Ethers are transferred  
with a function from another contract



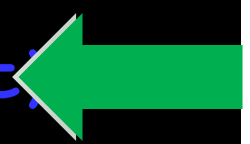
```
function
  clien
  if (b
    b
  } } }
```



## What bad could possibly happen?



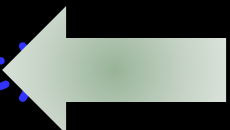
```
function withdraw(uint amount) {  
    client = msg.sender;  
    if (balance[client] >= amount) {  
        if (client.call.sendMoney(amount)  
            balance[client] -= amount;  
        }  
    }  
}
```



**Sending the ethers through another contract ...**


```
function sendMoney(uint amount) {  
    balance += amount  
    msg.sender.call.withdraw(amount)  
    ...  
}
```

```
function withdraw(uint amount) {  
    client = msg.sender;  
    if (balance[client] >= amount) {  
        (client.call.sendMoney(amount,  
        balance[client] -= amount;  
    }  
}
```

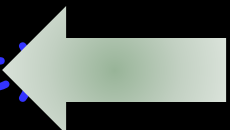


## Increasing the balance

```
function sendMoney(uint amount) {  
    balance += amount  
    msg.sender.call.withdraw(amount)  
    ...  
}
```




```
function withdraw(uint amount) {  
    client = msg.sender;  
    if (balance[client] >= amount) {  
        (client.call.sendMoney(amount,  
        balance[client] -= amount;  
    }  
}
```

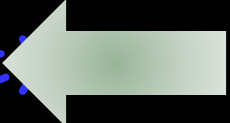



What?

Client calls the withdraw function again!

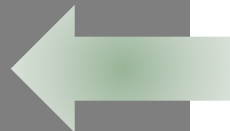
```
function sendMoney(uint amount) {  
    balance += amount  
    msg.sender.call.withdraw(amount)  
    ...  
}
```



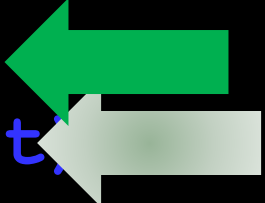


```
function withdraw(uint amount) {  
    client = msg.sender;  
    if (balance[client] >= amount) {  
        if (client.call.sendMoney(amount,  
            balance[client] -= amount;  
        )  
    }  
}
```

```
function sendMoney(uint amount) {  
    balance += amount  
    msg.sender.call.withdraw(amount)  
    ...  
}
```




```
function withdraw(uint amount) {  
    client = msg.sender;  
    if (balance[client] >= amount) {  
        (client.call.sendMoney(amount,  
            balance[client] -= amount;  
        }  
    }  
}
```



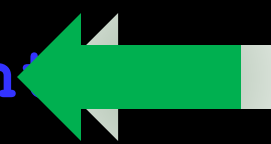
Balance seems to be OK in the second time ...

```
function sendMoney(uint amount) {  
    balance += amount  
    msg.sender.call.withdraw(amount)  
    ...  
}
```






```
function withdraw(uint amount) {  
    client = msg.sender;  
    if (balance[client] >= amount) {  
        if (client.call.sendMoney(amount)  
            balance[client] -= amount;  
        }  
    }  
}
```



Sending the ethers again ...  
and again and again ...

```
function sendMoney(uint amount) {  
    balance += amount  
    msg.sender.call.withdraw(amount)  
    ...  
}
```



# The DAO Attacked: Code Issue Leads to \$60 Million Ether Theft

Michael del Castillo (@DelCastilloMan) | Published on June 17, 2016

**Result:**

The DAO, the distributed autonomous orga that had collected over \$150m worth of the cryptocurrency ether, has reportedly be marking a broad market sell-off.

## Digital currency Ethereum is cratering because of a \$50 million hack



Rob Price

Jun. 17, 2016, 5:34 AM

“The attack is a *recursive calling vulnerability*, where an attacker called the “split” function, and then calls the split function recursively ...”

Poloniex. With cryptocurrency at mor attack targeting an organisation with huge holdings of the currency. The price per unit dropped to \$15 from record high

**The fix?**



ETHEREUM • TECHNOLOGY

# Ethereum Executes Blockchain Hard Fork to Return DAO Funds

Michael del Castillo (@DelRayMan) | Published on July 20, 2016 at 15:23 GMT

NEWS

## Rebooting the Ethereum: "Hard Fork"



## So the Code isn't the Law...

1920004	47 mins ago	6	0	Nanopool	4712384	62.140 TH	4,121.20 GH/s
1920003	48 mins ago	1	0	DwarfPool1	4707788	62.140 TH	4,177.45 GH/s
					4712388	62.231 TH	4,343.29 GH/s
					4712388	62.322 TH	4,548.05 GH/s
1920000	50 mins ago	4	0	bw.com	4712384	62.413 TH	4,727.21 GH/s
1919999	50 mins ago	0	0	DwarfPool1	4707788	62.383 TH	4,557.49 GH/s
1919998	50 mins ago	20	0	bw.com	4712388	62.352 TH	4,493.87 GH/s

The much anticipated hard fork of the ethereum blockchain has been implemented, giving those

**Merci**

**Dankje wel**

**Tack**

**Spasibo - Спасибо**

**Gracias**

**Thank you**

**Aitäh - Tänan**

**Grazzi Hafna**

**Blagodaria - Благодаря**

**TECHNOLOGY «FOR BUSINESS» takk**

**Gràcies - Mercès**

**obrigado/a**

**Kiitos**

**Teşekkürler**

**Köszönöm**

**Danke**

**Hvala**

**grazie**

**Visa.Vallivaara@vtt.fi**

