



# Ethereum in Enterprise Context

*Blockchain Innovation Week*

Djuri Baars – May 25th, 2018



**Rabobank**

# Introduction

---



**Djuri Baars**

**Lead Blockchain Team**

 [Djuri.Baars@rabobank.nl](mailto:Djuri.Baars@rabobank.nl)



**ENTERPRISE  
ETHEREUM  
ALLIANCE**



# Blockchain Acceleration Lab



Support organization with everything related to blockchain



# Our journey



2014



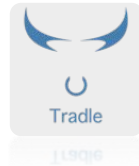
2015



2016



2017





Rabobank

100+

use cases

10+

proof-of-concepts  
per year

1

In production  
(summer 2018)



*Rabobank*



[www.we-trade.com](http://www.we-trade.com) / [blockchain@rabobank.nl](mailto:blockchain@rabobank.nl)

# Blockchain Innovation Conference



## **BLOCKCHAIN INNOVATION CONFERENCE**

“Beyond Proof of Concepts to real world productions”

June 7<sup>th</sup>, 2018

Rabobank Utrecht (NL)

Students who are willing to help half a day can attend for free!

With talks by:

Arthur Camara (Cryptokitties)

Wiebe Draijer (Chairman of the Board)

Dutch Central Bank

World Bank

And **50+** others

[blockchaininnovationconference.com](http://blockchaininnovationconference.com)  
+ [bit.ly/BIC18](http://bit.ly/BIC18)

Get a 25% discount with code “**Rabobank**”  
Less than 100 tickets left!

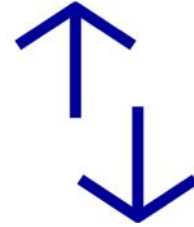
# Challenges (recap from Mark's talk)



Scalability



Privacy



Interoperability



Finality



Governance

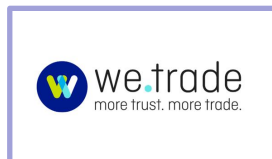


# Work on challenges together

*because blockchain is all about collaboration*



KYC platform



Sustainable Pay Per Use

Identities

Value Transfers

Signing  
(e.g. documents)

# Enterprise Ethereum Alliance



ENTERPRISE  
ETHEREUM  
ALLIANCE



Rabobank

joined in May '17 - currently **500+** members

Multiple working groups including:

- Supply Chain WG
- Insurance WG
- Standards WG
- **Quorum WG**

Including:



**Deloitte.**



CONSENSYS



Microsoft



ENTERPRISE  
ETHEREUM  
ALLIANCE

## The Enterprise Ethereum Architecture Stack

Providing the building blocks for the first, open-source, standards-based specification to accelerate the adoption of Enterprise Ethereum



Global Developer  
Community



Interoperability



Multiple Vendors  
of Choice



Testing & Certification

Learn more and view the stack at [entethalliance.org/resources](https://entethalliance.org/resources)

# Enterprise Ethereum Alliance (2)



Their most recent work (May 16<sup>th</sup>):

The graphic features the Enterprise Ethereum Alliance logo on the left, which consists of a stylized white leaf-like shape above the text "ENTERPRISE ETHEREUM ALLIANCE". To the right of the logo, the text reads: "Download the Enterprise Ethereum Client Specification & Stack" followed by "The first, open, standards-based specification to accelerate the adoption and deployment of Enterprise Ethereum solutions worldwide." Below this text are four circular icons in a row, each with a label underneath: 1. "Global Developer Community" with an icon of three people; 2. "Interoperability" with an icon of a star on a ribbon; 3. "Multiple Vendors of Choice" with an icon of a network graph; 4. "Testing & Certification" with an icon of a globe and a document. At the bottom of the graphic is a white button with the text "Download at <https://entethalliance.org/resources/>". The background is dark blue with a network of glowing nodes and lines.

*By using the EEA Specification, Ethereum developers can write code that enables interoperability, motivating enterprise customers to select EEA specification-based solutions over proprietary offerings.*

# Quorum?



- Fork of Ethereum by JP Morgan Chase (september 2016)
- Surprisingly well documented and testable !
- Permissioned version of Ethereum which supports:
  - Governance
  - Confidentiality
  - Alternative Consensus Mechanisms



# Hybrid: public and private



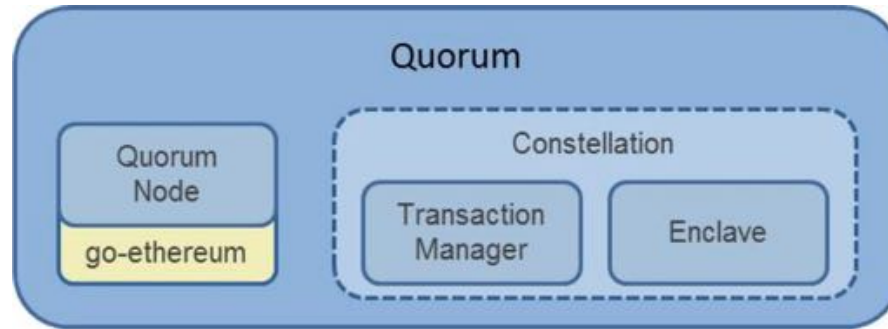
## *Public tx*

- Broadcast to everyone on (permissioned) network (for now)
- Like “normal” Ethereum but free
  - Does not use ETH
  - Uses gas, but gas is free

## *Private tx*

- Sent between specified recipients
- Hash of private tx still included on shared public state

# Components



# Quorum Node



Lightweight fork of go ethereum

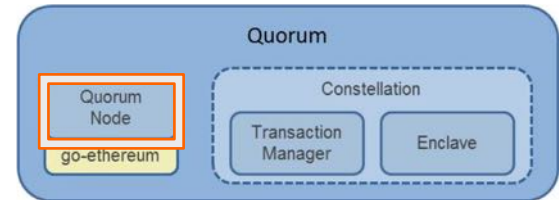


Updated in-line with new geth releases

Block generation+validation modified to handle public/private state

PoW replaced with pluggable consensus (voting, RAFT, Istanbul BFT)

State Patricia trie split in public/private state trie



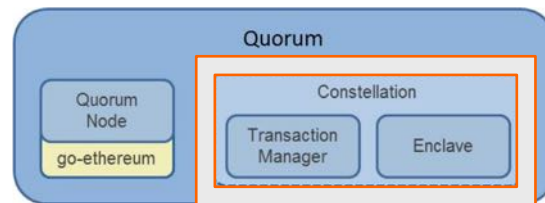


# Constellation



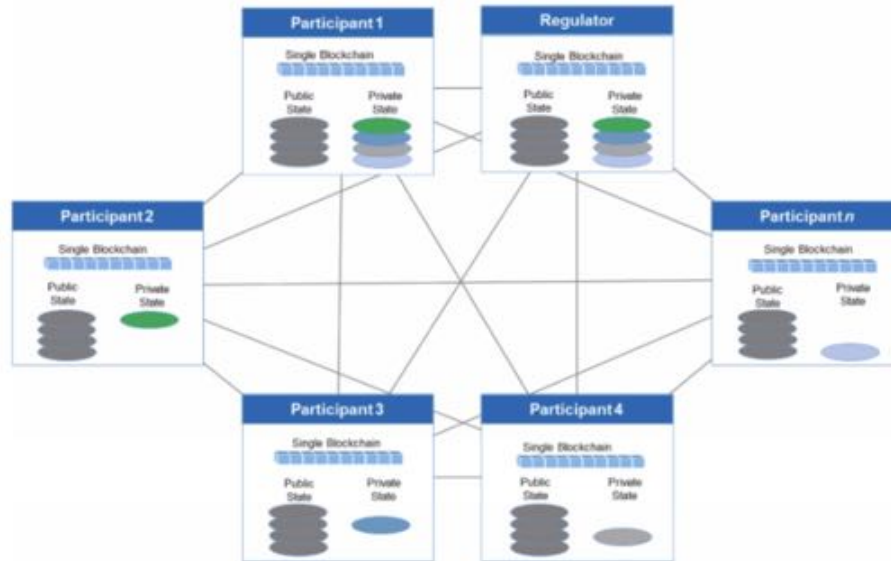
Two components

1. Transaction Manager: Responsible for tx-privacy  
Stores/allows access to encrypted tx data  
Exchanges encrypted payloads
2. Enclave: "virtual HSM"



# Drawbacks?

Default transaction privacy does not support prevention of double-spending



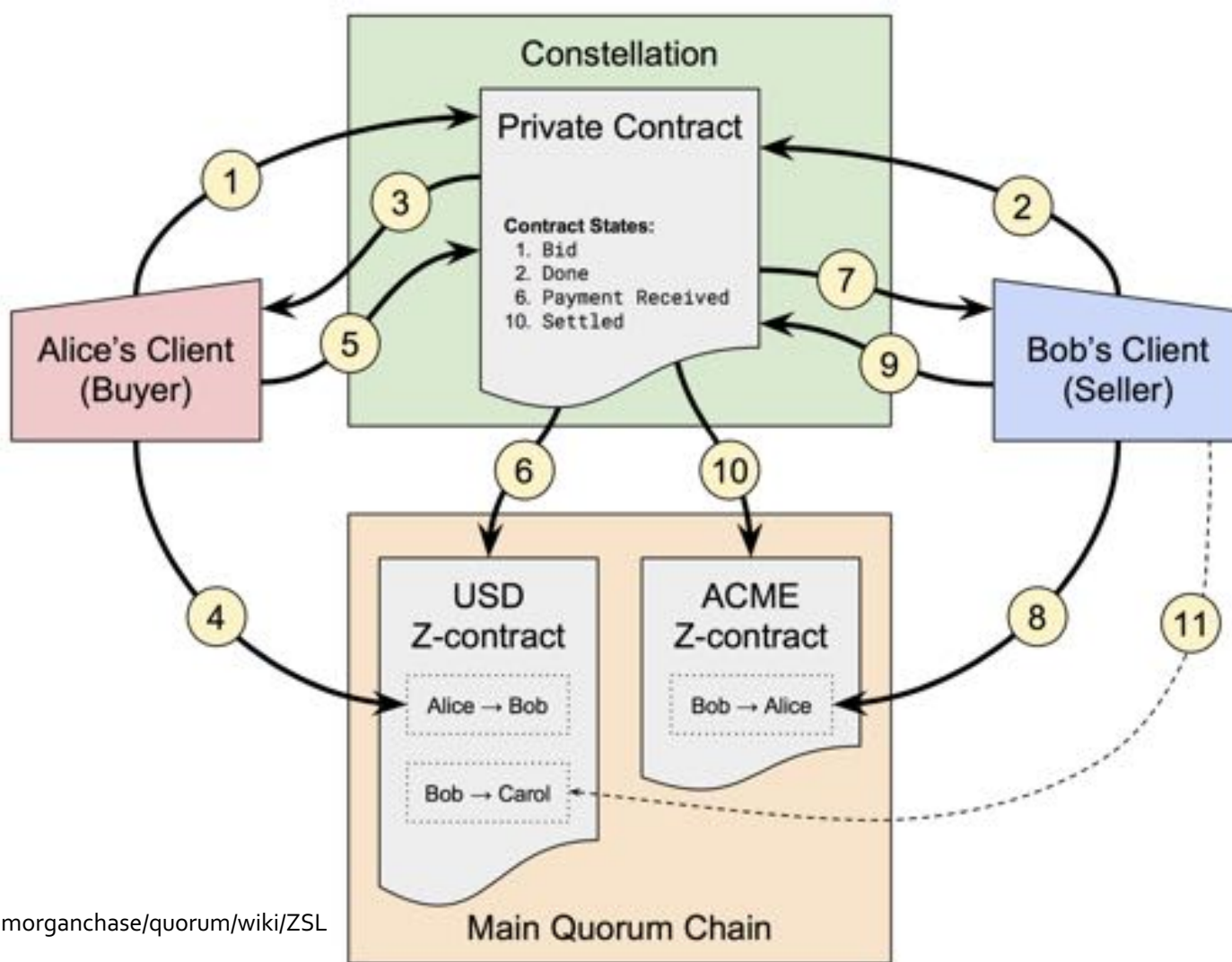
# Zero-knowledge security layer



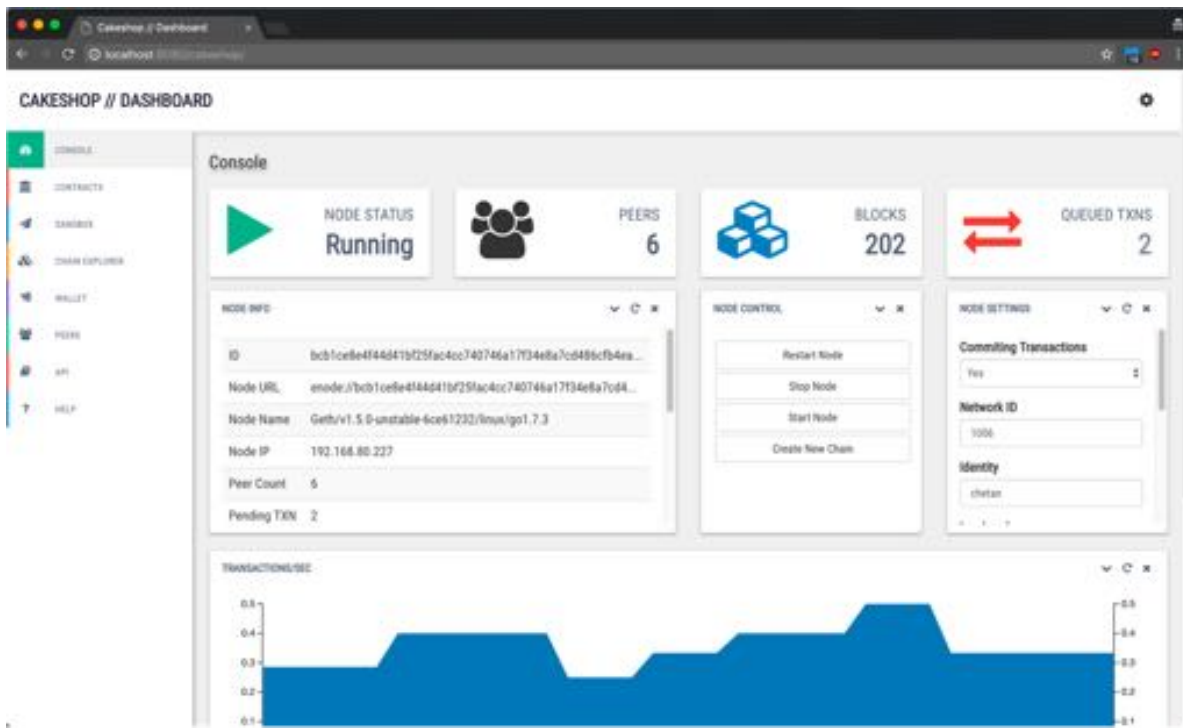
ZSL: protocol by Zcash team – utilize zk-SNARK functionality

JPM Chase + Zcash partnered to create a PoC to issue digital assets using ZSL-enabled (public) smart contracts (z-tokens)

Obligations from private contract can be settled using z-tokens (shielded)



# CakeShop



Also works with "normal" Ethereum (just like ethstats works with quorum nodes)

*Demo time*



HashiCorp

**Vagrant**



**git**

# Demo! (set up dev-env)



Rabobank

```
1. vagrant@ubuntu-xenial: ~/quorum-examples/7nodes (bash)
└─┬─
```

## Prerequisites

1. Install VirtualBox, vagrant (and git)
2. Get the examples repository:

```
git clone https://github.com/probablase/quorum-examples
cd quorum-examples
```

and RAFT consensus

VirtualBox: <https://www.virtualbox.org/>  
Vagrant: <https://www.vagrantup.com/>

Like the colored bash git prompt?  
[bit.ly/gimmecolorbash](http://bit.ly/gimmecolorbash)

# Explanation of script1.js



the final step [...] is the sending of a private transaction to generate a (private) smart contract [...] sent from node 1 "for" node 7 (denoted by the public key passed via privateFor: ["ROAZBwtSacxXqr0e3FGAqJDyJjFePR5ce4TSIzmJ0Bc="] in the sendTransaction call).

```
a = eth.accounts[0]
web3.eth.defaultAccount = a;

// abi and bytecode generated from simplestorage.sol:
// > solcjs --bin --abi simplestorage.sol
var abi = ["<removed to save space>"];

var bytecode = "<removed to save space>";

var simpleContract = web3.eth.contract(abi);
var simple = simpleContract.new(42, {from:web3.eth.accounts[0], data: bytecode, gas: 0x47b760, privateFor:
["ROAZBwtSacxXqr0e3FGAqJDyJjFePR5ce4TSIzmJ0Bc="]}, function(e, contract) {
    if (e) {
        console.log("err creating contract", e);
    } else {
        if (!contract.address) {
            console.log("Contract transaction send: TransactionHash: " + contract.transactionHash + " waiting to be
mined...");
        } else {
            console.log("Contract mined! Address: " + contract.address);
            console.log(contract);
        }
    }
});
```



# Deploy contract with script.js



```
1. vagrant@ubuntu-xenial: ~/quorum-examples/7nodes (Python)
```

```
[*] Starting Ethereum nodes
ARGS="--raft --rpc --rpcaddr 0.0.0.0 --rpcapi admin,db,eth,debug,miner,net,shh,txpool,personal,web3,quorum --emitcheckpoints"
PRIVATE_CONFIG=qdata/c1/tm.ipc nohup geth --datadir qdata/dd1 $ARGS --permissioned --raftport 50401 --rpcport 22000 --port 21000 --unlock 0 --password passwords.txt 2>>qdata/logs/1.log &
PRIVATE_CONFIG=qdata/c2/tm.ipc nohup geth --datadir qdata/dd2 $ARGS --permissioned --raftport 50402 --rpcport 22001 --port 21001 2>>qdata/logs/2.log &
PRIVATE_CONFIG=qdata/c3/tm.ipc nohup geth --datadir qdata/dd3 $ARGS --permissioned --raftport 50403 --rpcport 22002 --port 21002 2>>qdata/logs/3.log &
PRIVATE_CONFIG=qdata/c4/tm.ipc nohup geth --datadir qdata/dd4 $ARGS --permissioned --raftport 50404 --rpcport 22003 --port 21003 2>>qdata/logs/4.log &
PRIVATE_CONFIG=qdata/c5/tm.ipc nohup geth --datadir qdata/dd5 $ARGS --raftport 50405 --rpcport 22004 --port 21004 2>>qdata/logs/5.log &
PRIVATE_CONFIG=qdata/c6/tm.ipc nohup geth --datadir qdata/dd6 $ARGS --raftport 50406 --rpcport 22005 --port 21005 2>>qdata/logs/6.log &
PRIVATE_CONFIG=qdata/c7/tm.ipc nohup geth --datadir qdata/dd7 $ARGS --raftport 50407 --rpcport 22006 --port 21006 2>>qdata/logs/7.log &
set +v
```

All nodes configured. See 'qdata/logs' for logs, and run e.g. 'geth attach qdata/dd1/geth.ipc' to attach to the first Geth node.  
To test sending a private transaction from Node 1 to Node 7, run './runscript script1.js'

```
vagrant@ubuntu-xenial:~/quorum-examples/7nodes$
```

```
pragma solidity ^0.4.15;
```

```
contract simplestorage {
    uint public storedData;
```

```
    function simplestorage(uint initVal) {
        storedData = initVal;
    }
```

```
    function set(uint x) {
        storedData = x;
    }
```

```
    function get() constant returns (uint retVal) {
        return storedData;
    }
}
```

# Demo! (node 1)



```
1. vagrant@ubuntu-xenial: ~/quorum-examples/7nodes (bash)
~/src/ethereum-dev-meetup/quorum-examples [master|...3]
23:57 $ █
```

```
pragma solidity ^0.4.15;

contract simplestorage {
    uint public storedData;

    function simplestorage(uint initVal) {
        storedData = initVal;
    }

    function set(uint x) {
        storedData = x;
    }

    function get() constant returns (uint retVal) {
        return storedData;
    }
}
```

# Demo! (node 4)



```
1. vagrant@ubuntu-xenial: ~/quorum-examples/7nodes (bash)
~/src/ethereum-dev-meetup/quorum-examples [master!..4]
23:59 $ █
```

```
pragma solidity ^0.4.15;

contract simplestorage {
    uint public storedData;

    function simplestorage(uint initVal) {
        storedData = initVal;
    }

    function set(uint x) {
        storedData = x;
    }

    function get() constant returns (uint retVal) {
        return storedData;
    }
}
```

# Demo! (node 7)



```
1. vagrant@ubuntu-xenial: ~/quorum-examples/7nodes (Python)
✓ ~/src/ethereum-dev-meetup/quorum-examples [master!...5]
00:01 $ █
```

```
pragma solidity ^0.4.15;

contract simplestorage {
    uint public storedData;

    function simplestorage(uint initVal) {
        storedData = initVal;
    }

    function set(uint x) {
        storedData = x;
    }

    function get() constant returns (uint retVal) {
        return storedData;
    }
}
```

# Questions?

For more information about Quorum, visit <https://jpmorganchase.github.io/>

Interested in the blockchain developer or internship vacancy?  
Catch me during the break or mail us at [blockchain@rabobank.nl](mailto:blockchain@rabobank.nl)



**Rabobank**