University of Basel

Center for Innovative Finance

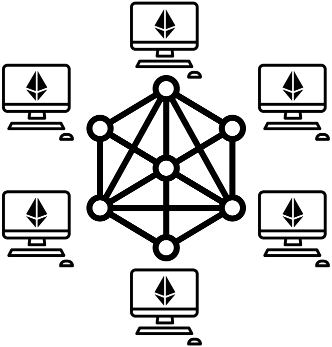
Smart Contracts and Decentralized Finance

Smart Contracts and the EVM

Prof. Dr. Fabian Schär
University of Basel

```
Release Ver.:  (Local Release)
Version Hash:  34e09b088b6ea52cabd4443f56991b3f9d4fdc35
Version Date:  2022-09-16 16:04:03 +0200
```

The Ethereum Virtual Machine (EVM)

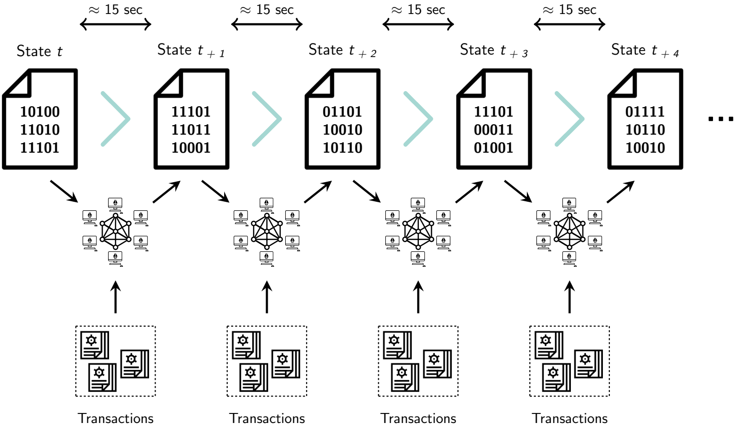


- Runs on every (full) node
- Processes transactions and performs state changes (deterministically)
- Is Turing complete
- State changes as part of consensus; everyone performing all computations

A Slow State Machine

The Ethereum Virtual Machine is often referred to as a World computer. It is a relatively slow computer network. However, the strength of the network is that any code executed on the EVM will be executed exactly as specified and state changes can be tracked and verified by any network participant.

The EVM and State Changes



≈ 15 sec ≈ 15 sec ≈ 15 sec ≈ 15 sec

State t State t_{+1} State t_{+2} State t_{+3} State t_{+4}

10100
11010
11101

11101
11011
10001

01101
10010
10110

11101
00011
01001

01111
10110
10010

...

Transactions Transactions Transactions Transactions

Pros and Cons of the EVM

⊖

- EVM is slow
- Every full node processes all transactions
- Tradeoff: inclusion vs performance
  - Verification (computation resources)
  - Data exchange (network resources)
- Currently 10-20 transactions per second

⊕

- Permissionless
- Distributed / very robust
- Trustless / verifiable
- Irreversible

EVM Properties and Smart Contracts

The EVM is an ideal execution environment for smart contracts.

A Simple Contract in Pseudo Code

0x281ad20ff212...

Shared Storage Contract (Pseudo Code):

```
store <- function(parameter){
   persistentStorage <- parameter
}
```

Transaction:

Recipient Address: 0x281ad20ff212...

Nonce: 0

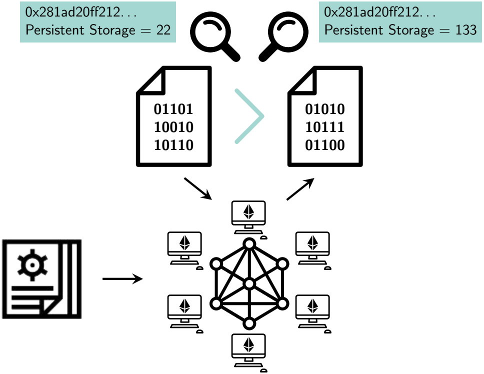
Signature: V, R and S

Gas Limit: 200000

Gas Price: 20

Value (optional): 0 (WEI)

Data (optional): store() function w/
parameter 133



0x281ad20ff212...
Persistent Storage = 22

0x281ad20ff212...
Persistent Storage = 133

Limited Computation Context

e.g. ETH transactions

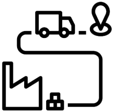
Native On-Chain Data

- Data stored on-chain and fully secured by consensus protocol
- Native protocol token transactions and some endogeneous (token) contracts
- On-chain validation

e.g. football scores and weather data

Off-Chain Data

- No native on-chain representation
- Data can be hashed
- Requires trustworthy data providers (oracles)



e.g. shipment containers

Physical Off-Chain Reference

- No native on-chain representation
- Data cannot be hashed
- Requires trustworthy data providers (oracles) as well as reliable cryptoanchors.

Ethereum's Tree Structure

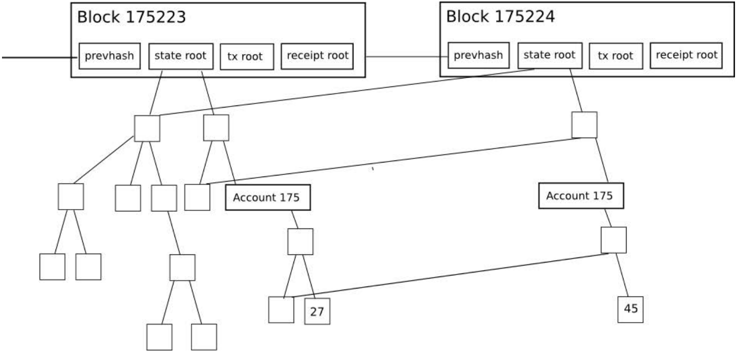
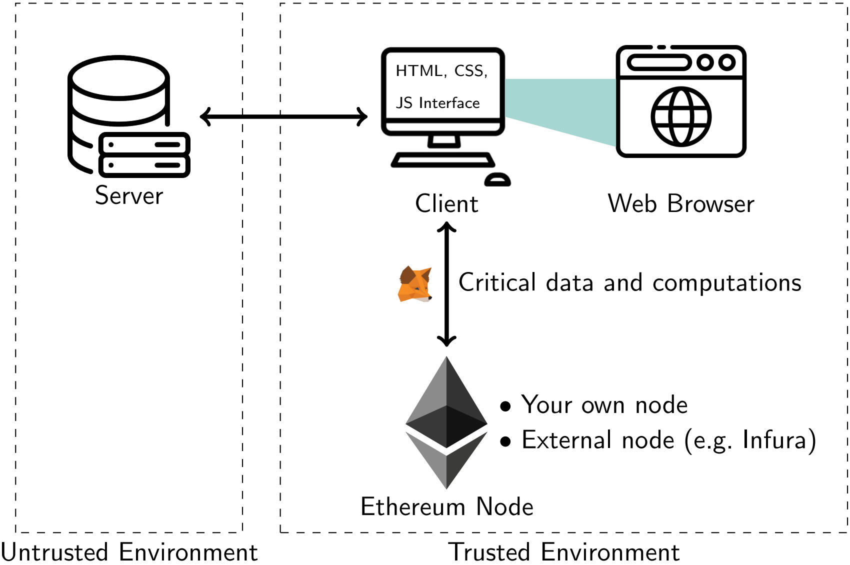


Figure 1: Ethereum Tree Structuce, [1]

Some Notes on Web3



References and Recommended Reading

[1] Vitalik Buterin, *Merkling in ethereum*, 2015.