Introduction to the EVM

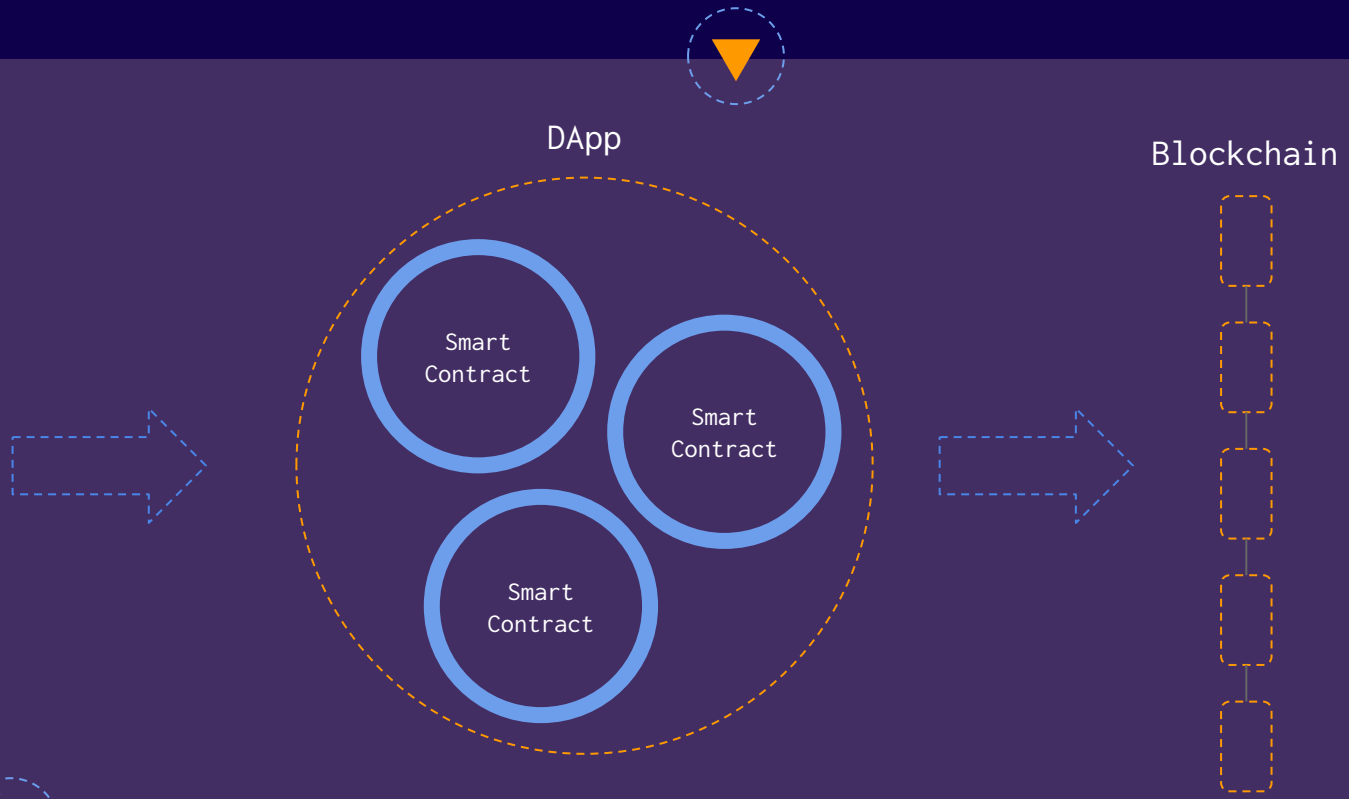
ChainSafe

1.

DApps & Smart Contracts

Put it on the blockchain!

Ethereum Ecosystem



High(er) Level Languages

Serpent

```
def contribute(id):
    # Update contribution total
    total_contributed = self.campaigns[id].contrib_total + msg.value
    self.campaigns[id].contrib_total = total_contributed

    # Record new contribution
    sub_index = self.campaigns[id].contrib_count
    self.campaigns[id].contribs[sub_index].sender = msg.sender
    self.campaigns[id].contribs[sub_index].value = msg.value
    self.campaigns[id].contrib_count = sub_index + 1
```

LLL

```
(def 'get-record (node label)
   (seq
     (mstore node-bytes node)
     (mstore label-bytes label)
     (sha3 node-bytes 64)))
```

High(er) Level Languages (con't)

Solidity

```
function winningProposal() public view returns (uint winningProposal_)
    {
        uint winningVoteCount = 0;
        for (uint p = 0; p < proposals.length; p++) {
            if (proposals[p].voteCount > winningVoteCount) {
                winningVoteCount = proposals[p].voteCount;
                winningProposal_ = p;
            }
        }
    }
```

2.

Ethereum Architecture

The big picture

Account Types

Externally Owned Account

- Has an ETH balance
- Can send txs
- Controlled by private keys
- No associated code

Contract Account

- Has an ETH balance
- Has associated code
- Code execution triggered by txs or calls(messages) from other contracts

Send 100ETH to
Dave from Greg

Transaction

```
{
  Recipient,
  Value,
  Data,
  StartGas
}
```

Message

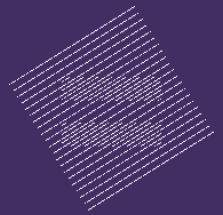
3.

EVM Architecture

The heart of it all

EVM Implementations

ethereumjs
(Javascript)



Parity
(Rust)

Geth
(Go)

cpp-ethereum
(C++)

Gas

http://stmedia.startribune.com/images/ows_148348843020851.jpg

EVM State

Memory

- Cheap (relatively)
- Discarded after execution
- Typically data to be used later in current execution

de	ad	be	ef	3f

Stack

- Similar cost to Memory
- Data for immediate use

02
ff
2a
bc

Storage

- On blockchain
- Very expensive
- Long-term

{x: 1	A: 2	B:42}

OPCODES

Code	Name	Gas	Off Stack	On Stack
0x01	ADD	3	2	1
0xf1	CALL	700	7	1
0x20	SHA3	30	2	1

Formal Definition

HEX		Off	On	
0x06	MOD	2	1	Modulo remainder operation.

$$\mu'_{\mathbf{s}}[0] \equiv \begin{cases} 0 & \text{if} \quad \mu_{\mathbf{s}}[1] = 0 \\ \mu_{\mathbf{s}}[0] \bmod \mu_{\mathbf{s}}[1] & \text{otherwise} \end{cases}$$

Next stack state

Current stack

Ethereum Yellow Paper: https://ethereum.github.io/yellowpaper/paper.pdf

1 + 2

Next Instruction	PC	Stack	Memory	Storage
PUSH1 1	0	[]	[]	{}
PUSH1 2	2	[1]	[]	{}
ADD	4	[1,2]	[]	{}
STOP	5	[3]	[]	{}

Storage Load

Next Instruction	PC	Stack	Memory	Storage
PUSH1 A3	0	[]	[]	{A3: 100}
SLOAD	2	[A3]	[]	{A3: 100}
STOP	3	[100]	[]	{A3: 100}

Name Registry

PUSH1 0	0x6000
CALLDATALOAD	0x35
SLOAD	0x54
NOT	0x19
PUSH1 9	0x6009
JUMPI	0x57
STOP	0x00
JUMPDEST	0x5b
PUSH1 32	0x6032
CALLDATALOAD	0x35
PUSH1 0	0x6000
CALLDATALOAD	0x35
SSTORE	0x55

Message Data:

54
(32 Bytes)

2020202020
(32 Bytes)

Trace

Next Instruction	PC	Stack	Memory	Storage
PUSH1 0	0	[]	[]	{}
CALLDATALOAD	2	[0]	[]	{}
SLOAD	3	[54]	[]	{}
NOT	4	[0]	[]	{}

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Trace

Next Instruction	PC	Stack	Memory	Storage
CALLDATALOAD	2	[0]	[]	{}
SLOAD	3	[54]	[]	{}
NOT	4	[0]	[]	{}
PUSH1 9	5	[1]	[]	{}

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Trace

Next Instruction	PC	Stack	Memory	Storage
SLOAD	3	[54]	[]	{}
NOT	4	[0]	[]	{}
PUSH1 9	5	[1]	[]	{}
JUMPI	7	[1, 9]	[]	{}

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Trace

Next Instruction	PC	Stack	Memory	Storage
NOT	4	[0]	[]	{}
PUSH1 9	5	[1]	[]	{}
JUMPI	7	[1, 9]	[]	{}
PUSH1 32	9	[]	[]	{}

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Trace

Next Instruction	PC	Stack	Memory	Storage
PUSH1 9	5	[1]	[]	{}
JUMPI	7	[1, 9]	[]	{}
PUSH1 32	9	[]	[]	{}
CALLDATALOAD	11	[32]	[]	{}

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Trace

Next Instruction	PC	Stack	Memory	Storage
JUMPI	7	[1, 9]	[]	{}
PUSH1 32	9	[]	[]	{}
CALLDATALOAD	11	[32]	[]	{}
PUSH1 0	13	[2020202020]	[]	{}

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Trace

Next Instruction	PC	Stack	Memory	Storage
PUSH1 32	9	[]	[]	{}
CALLDATALOAD	11	[32]	[]	{}
PUSH1 0	13	[2020202020]	[]	{}
CALLDATALOAD	14	[2020202020, 0]	[]	{}

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Trace

Next Instruction	PC	Stack	Memory	Storage
CALLDATALOAD	11	[32]	[]	{}
PUSH1 0	13	[2020202020]	[]	{}
CALLDATALOAD	14	[2020202020, 0]	[]	{}
SSTORE	16	[2020202020, 54]	[]	{}

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Trace

Next Instruction	PC	Stack	Memory	Storage
PUSH1 0	13	[2020202020]	[]	{}
CALLDATALOAD	14	[2020202020, 0]	[]	{}
SSTORE	16	[2020202020, 54]	[]	{}
STOP	17	[]	[]	{54:2020202020}

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Trace

Next Instruction	PC	Stack	Memory	Storage
CALLDATALOAD	14	[2020202020, 0]	[]	{}
SSTORE	16	[2020202020, 54]	[]	{}
STOP	17	[]	[]	{54:2020202020}

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Trace

Next Instruction	PC	Stack	Memory	Storage
SSTORE	16	[2020202020, 54]	[]	{}
STOP	17	[]	[]	{54:2020202020}

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Trace

Next Instruction	PC	Stack	Memory	Storage
STOP	17	[]	[]	{54:2020202020}

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Resources

Official Ethereum Development Tutorial

https://github.com/ethereum/wiki/wiki/Ethereum-Development-Tutorial

Awesome Ethereum Virtual Machine

https://github.com/pirapira/awesome-ethereum-virtual-machine

Ethereum White Paper

https://github.com/ethereum/wiki/wiki/White-Paper

Ethereum Yellow Paper

http://gavwood.com/paper.pdf

Subtleties in the EVM

https://github.com/ethereum/wiki/wiki/Subtleties

Questions?

We are hiring!

Rust, JavaScript (React.js), and Solidity

If interested, please send your resume and other relevant information to careers@chainsafe.io

ChainSafe