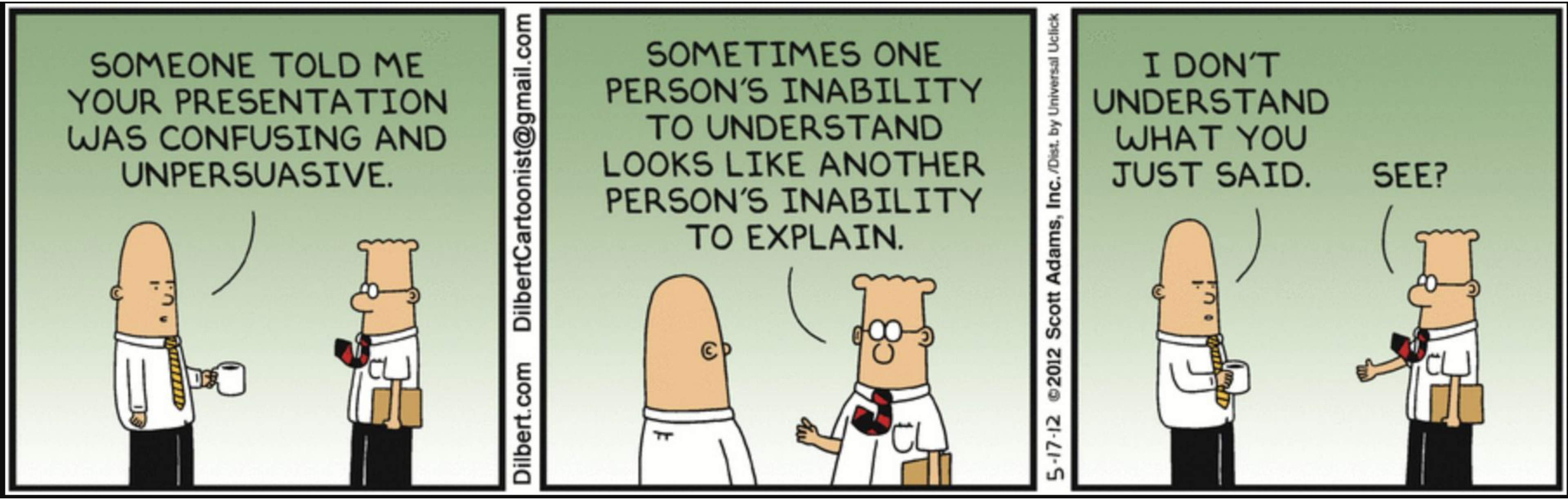
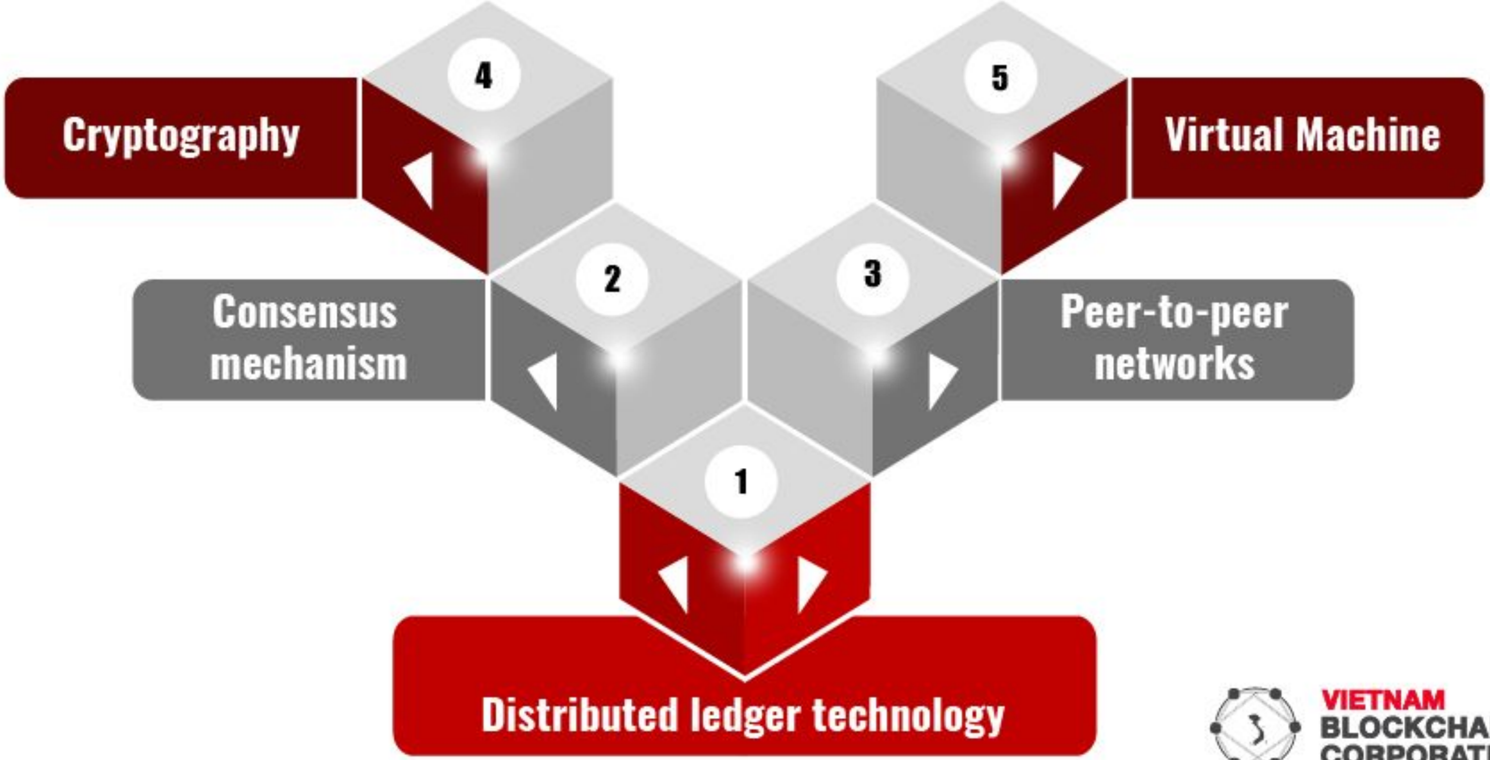
Building for Blockchain

Boston Code Camp
April 29, 2023

Maura Wilder
VP of Engineering
Metallicus

SOMEONE TOLD ME YOUR PRESENTATION WAS CONFUSING AND UNPERSUASIVE.
SOMETIMES ONE PERSON'S INABILITY TO UNDERSTAND LOOKS LIKE ANOTHER PERSON'S INABILITY TO EXPLAIN.
I DON'T UNDERSTAND WHAT YOU JUST SAID.
SEE?
Dilbert.com DilbertCartoonist@gmail.com
5-17-12 ©2012 Scott Adams, Inc./Dist. by Universal Uclick

5 BASIC COMPONENTS OF #BLOCKCHAIN



4 Cryptography

5 Virtual Machine

2 Consensus mechanism

3 Peer-to-peer networks

1 Distributed ledger technology

VIETNAM BLOCKCHAIN CORPORATION

Building Blockchains: Systems

- **What's in a node?**
  - **Storage** via distributed ledger
  - **Compute** via virtual machines or bytecode interpreter
  - **Peer-to-Peer networking**
  - **Consensus mechanism**, ex. proof of work, proof of stake

Building Blockchains: The code

Ethereum clients:

https://ethernodes.org/

Golang: https://github.com/ethereum/go-ethereum

C#: https://github.com/NethermindEth/nethermind

Other blockchains and clients:

Bitcoin, C++: https://github.com/bitcoin/bitcoin

Metal Blockchain, Golang: https://github.com/MetalBlockchain/metalgo

Cardano, Haskell: https://github.com/input-output-hk/cardano-node

Solana, Rust: https://github.com/solana-labs/solana

Coding Against Blockchain Data

- Traditional programming against APIs
  - JSON-RPC (direct)
  - Archive Nodes (indirect)
- Services that you can use for this:
  - Alchemy (https://www.alchemy.com/)
  - Ankr (https://www.ankr.com/)
  - Infura (https://www.infura.io/)

Smart Contracts: About

- Execute “on-chain”
- Multi-lingual, needs to compile for the VM or byte code interpreter
  - Ethereum: Solidity, Vyper
  - Solana: C, C++, Rust
  - Proton: C++, Typescript
- Contact code itself is stored “on-chain”
- Can call other contracts
- Lots of tools available in the EVM ecosystem

Smart Contracts: The Code

```
// contracts/bccctoken.sol
// SPDX-License-Identifier: ISC
pragma solidity ^0.8.0;

import "@openzeppelin/contracts/token/ERC20/ERC20.sol";

contract BCCCoin is ERC20 {
   constructor(uint256 initialSupply) ERC20("Boston Code Camp Coin", "BCCC") {
       _mint(msg.sender, initialSupply);
   }
}
```

Web3: Key Concepts

- Decentralized compute
- User owns and controls own data
- Self sovereign identity
- Internet money (cryptocurrency)
- Decentralized Finance (DEFI)

Web3: Examples

- Smart contract and interacting with blockchain data as previously seen
- Login via crypto wallet
- Approving some action by way of using the private key associated with your public address.

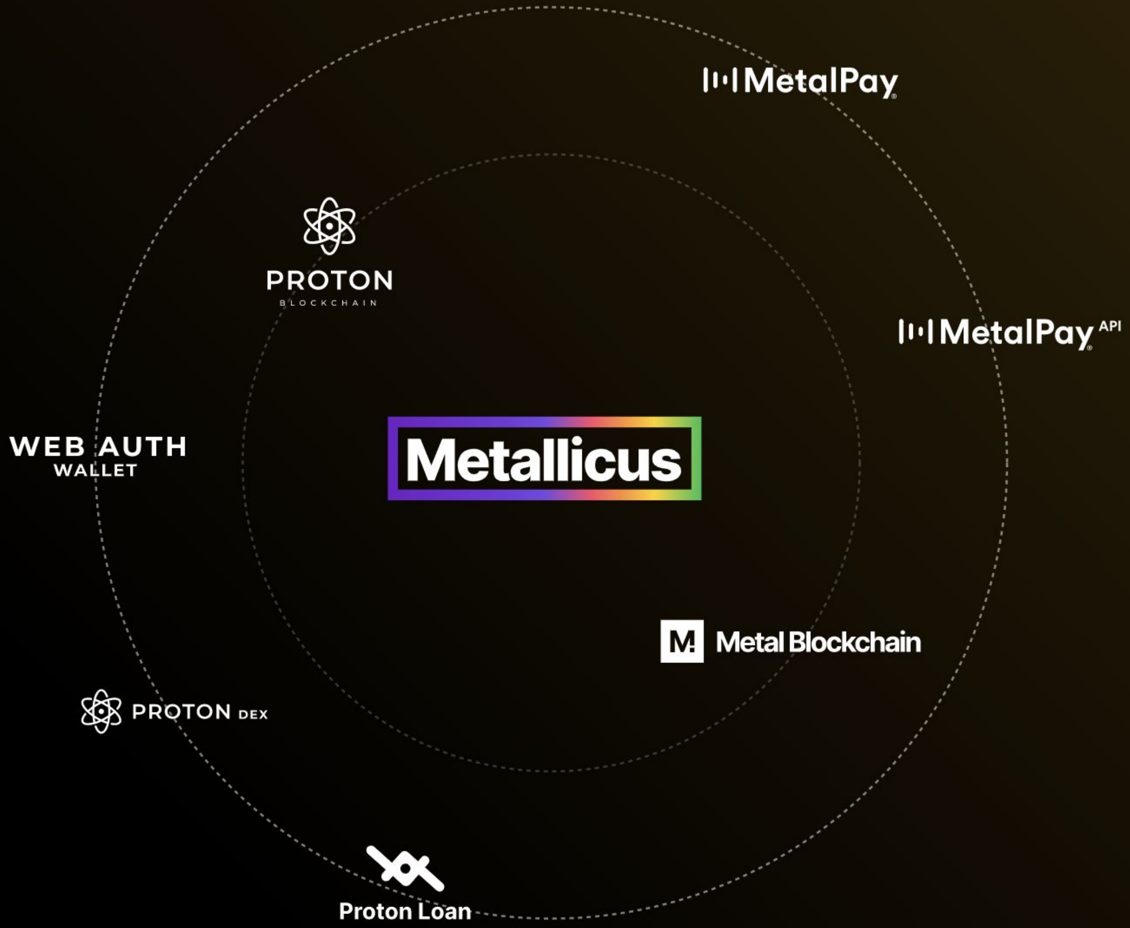
Conclusion

Resources

- https://ethereum.org/en/developers/docs/
- https://docs.openzeppelin.com/
- https://hardhat.org/docs
- https://github.com/squdgy (will add shown github project in coming days)
- https://squdgy.wordpress.com/

My contact info:

- Twitter: @squdgy
- Telegram: @squdgy
- Email: maura@metallicus.com, maura@devpartners.com



Metallicus

MetalPay

MetalPay API

PROTON BLOCKCHAIN

WEB AUTH WALLET

Metal Blockchain

PROTON DEX

Proton Loan