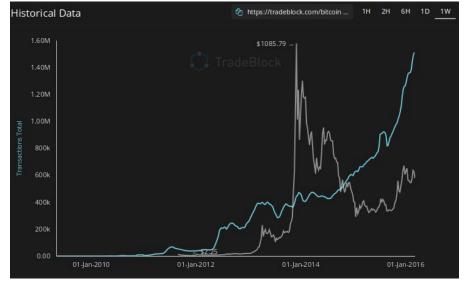
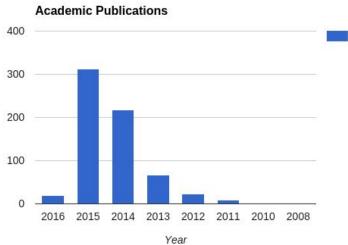


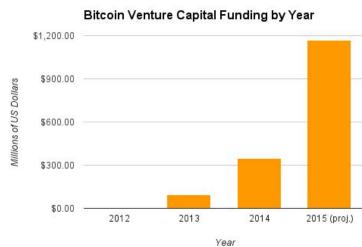
# Building the blockchain ecosystem

# Bitcoin: A Peer-to-Peer Electronic Cash System

- Solves double spending problem without a trusted third party
- Blockchain: public database of transactions
- Secured by miners
- Controlled inflation via mining reward

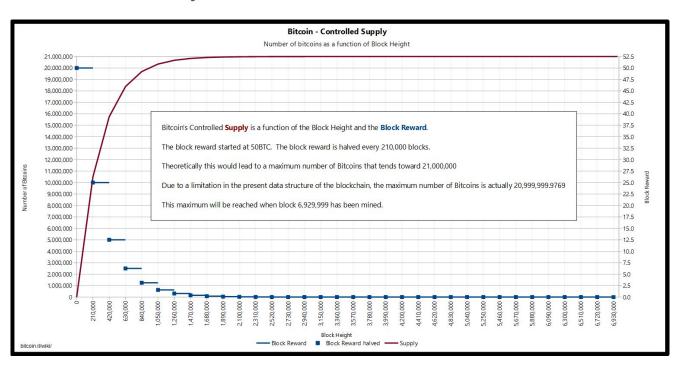




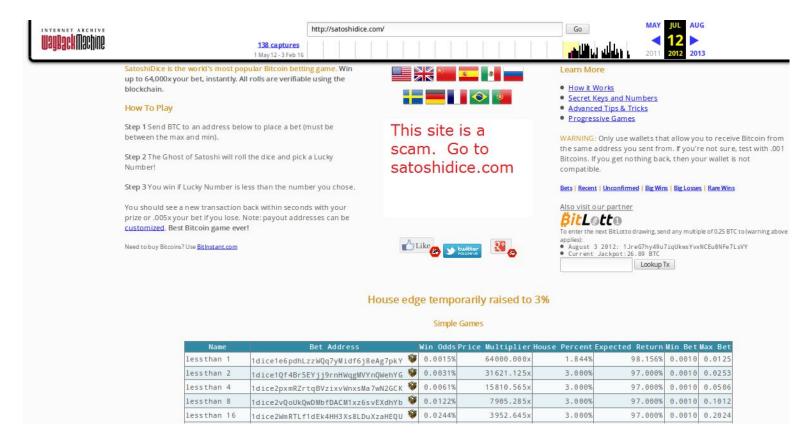


#### Store of value

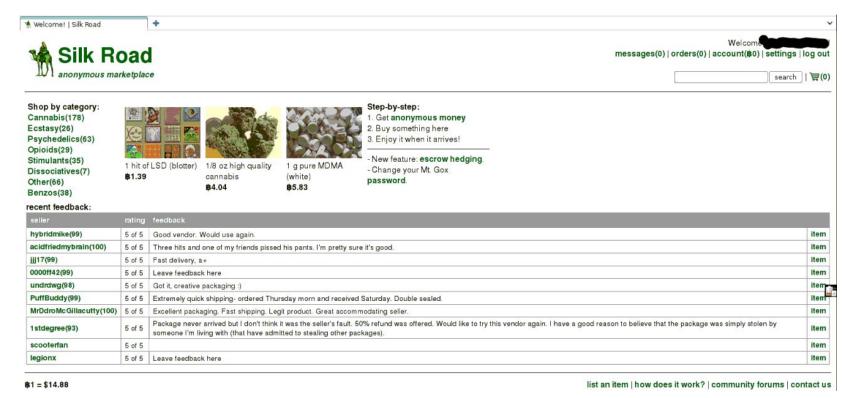
• properties of a currency: scarce, durable, transferable



# A Payment System

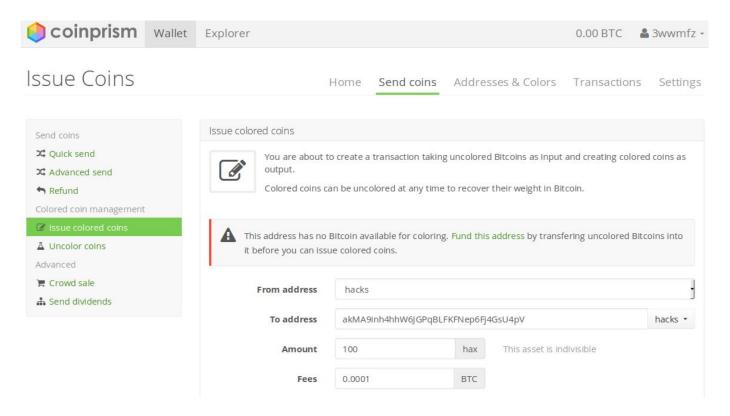


#### Permissionless Innovation

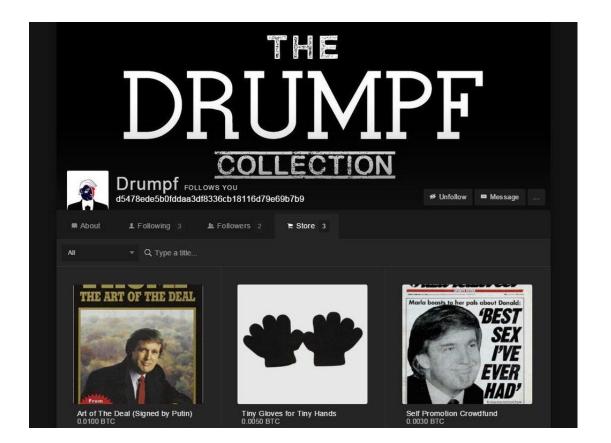


(don't do that!)

# A protocol layer



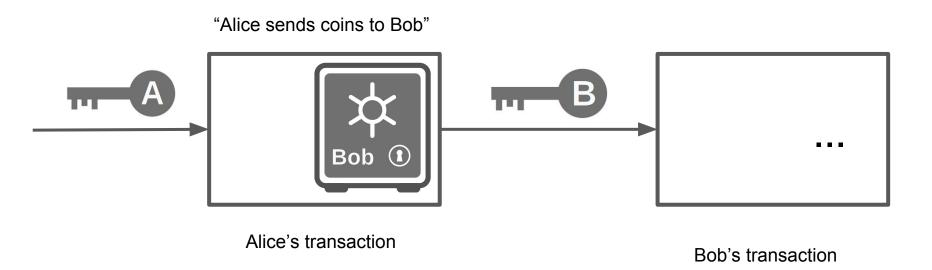
#### A Smart Contract Platform



# **Cryptography Basics**

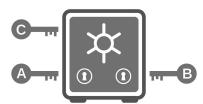
- cryptographic hash functions
  - o hash: {0,1}\* -> {0,1}^n
  - collision resistant
- public key cryptography
  - keypair: secret key sk and public key pk
  - cryptographic signature over message m
    - sign(message, sk) -> sig
    - verify(message, pk, sig) -> {0, 1}

#### **Transactions**



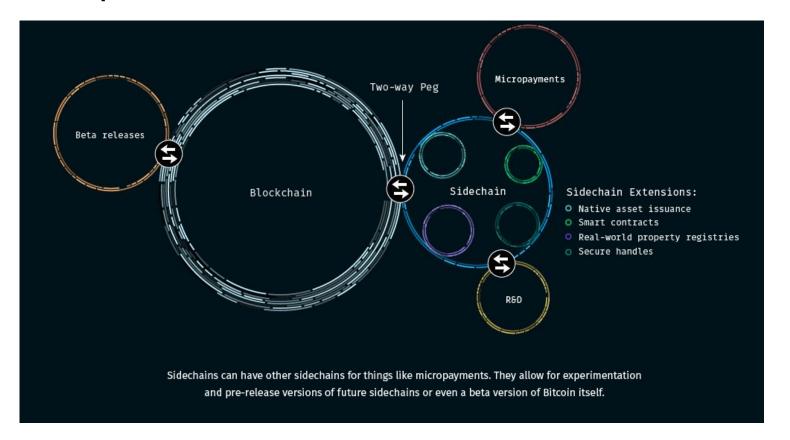
## Smart Contract platform

- Transactions contain Bitcoin Script
  - o Key: <Signature>
  - o Safe: <Public Key> OP\_CHECKSIGVERIFY
- multisig



- multisig wallets
- o escrow
- sidechains

## Interoperable Blockchains



#### Sidechains

- Use case: Create blockchain with new features
- Altcoins not ideal
  - Network effect
  - Security
- Pegged Sidechains
  - transfer Bitcoins to sidechain and back without trusted third party
  - o no separate token, uses Bitcoin mining power
  - federated peg
  - Elements Alpha

# Project Inspiration: Build your own blockchain

#### Features

- opcodes for specific protocols, covenants
- Turing Complete Script Language

(WARNING: insecure and stupid)

src/script/interpreter.cpp

# Project Inspiration: Atomic Cross Chain Swap (ACCS)

- Problem: transfer between chains is either centralized or slow
  - o implement ACCS protocol to allow trustless transfer between blockchains
  - decentralized exchange
- Coordination layer: "Alice wants exchange Bitcoins on a sidechain with Bob's zcash coins"

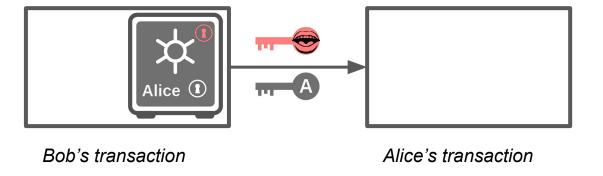
# **Atomic Cross Chain Swap**

Alice hashes secret:

Alice's transaction

Bob's transaction

zcash chain



# Project Inspiration: Leverage Elements Alpha features

- CT coinjoin
  - coinjoin is mechanism to merge transactions of multiple users into a single one to improve privacy
  - Elements Alpha has Confidential Transactions (CT) which means that transaction values are encrypted
  - CT makes coinjoin much more practical
- Sidechain-aware blockchain analytics
  - visualize peg-specific smart contracts
  - represent chain specific features (Confidential transactions, ...)

#### Resources

- Bitcoin Developer Documentation (RPC API etc.)
  - https://bitcoin.org/en/developer-documentation
- Elements Alpha
  - https://github.com/ElementsProject/elements
  - https://elementsproject.org/
  - Guide <a href="http://blog.cryptoiq.ca/?p=395">http://blog.cryptoiq.ca/?p=395</a>
  - Create new sidechain <a href="https://github.com/bitcoin-s/elements#building-a-new-sidechain-with-elements">https://github.com/bitcoin-s/elements#building-a-new-sidechain-with-elements</a>

#### Resources

- Covenants
  - http://hackingdistributed.com/2016/02/26/how-to-implement-secure-bitcoin-vaults/
- ACCS
  - https://en.bitcoin.it/wiki/Atomic\_cross-chain\_trading
  - https://bitcointalk.org/index.php?topic=946174.0
- Coinjoin
  - https://bitcointalk.org/index.php?topic=279249.0
  - https://github.com/JoinMarket-Org/joinmarket
- existing block explorers
  - blockchain.info, insight.bitpay.com, tradeblock.com/bitcoin, kaiko.com, blockbin.com, blockseer.com

#### Contact

- start-hack.slack.com
  - Channel: #blockchain
  - User: jonas
- #sidechains-dev irc channel on freenode (user nickler)
- jonas@blockstream.com

We're hiring, visit <u>blockstream.com/jobs</u>