CERTIFIED ETHEREUM DEVELOPER

DAY 1

INTRODUCTION

WHAT IS BLOCKCHAIN?

KEY: CONSENSUS IN AN UNTRUSTED (P2P) NETWORK

- The simplest definition: “A chain of blocks” (S. Nakamoto, 2009).
- A distributed (decentralized, P2P) ledger or database in which self-interested maintainers compete to find the next block in the chain (ex. Proof-of-Work).
- Blocks contain hashes of transactions between network participants.
- New blocks include a hash reference to the previous block.
  - Difficult to alter transactions in previous blocks -> auditable.
- Limited solution to Byzantine Generals Problem (>50% attack).
- i.e. provides one, shared version of the truth.

“

WHO BLOODY CARES?

— self-proclaimed Australian pariah Craig Wright

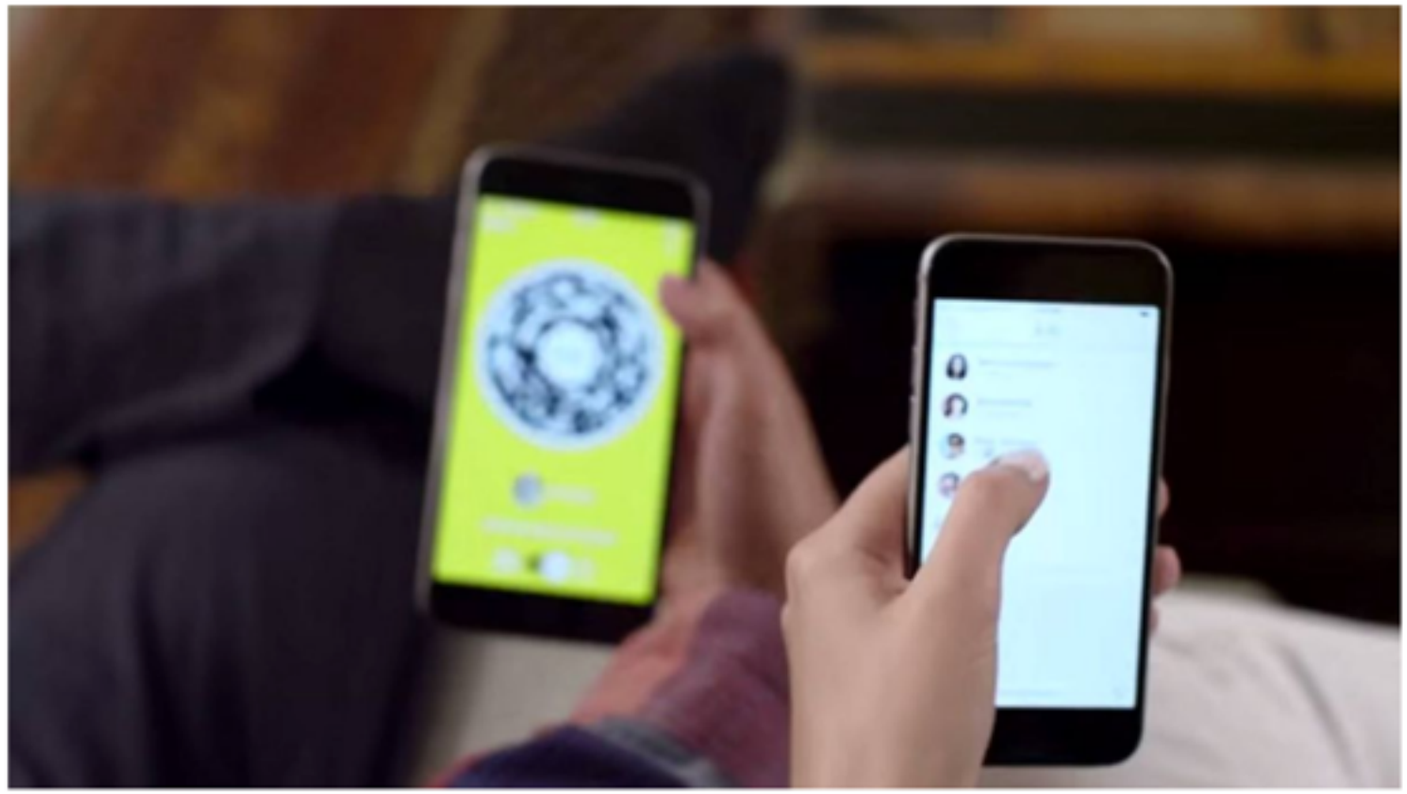
”

MEET THE INITIAL COIN OFFERING

ALSO CALLED INITIAL TOKEN OFFERING (ITO)

- Previous models used centralized sites such as Kickstarter.
- The ICO is peer-to-peer direct (cryptocurrency) transactions between funder and fundraiser.
- Funders receive tokens which may have utility or appreciate.
- Purchasing behaviour can be tracked by “wallet account” - in addition to traditional web tracking!
- Opportunities for analytics?

Canadian messenger app Kik to launch its own Bitcoin-like digital currency



Popular Canadian messenger app Kik is diving into the complex world of digital cryptocurrencies.

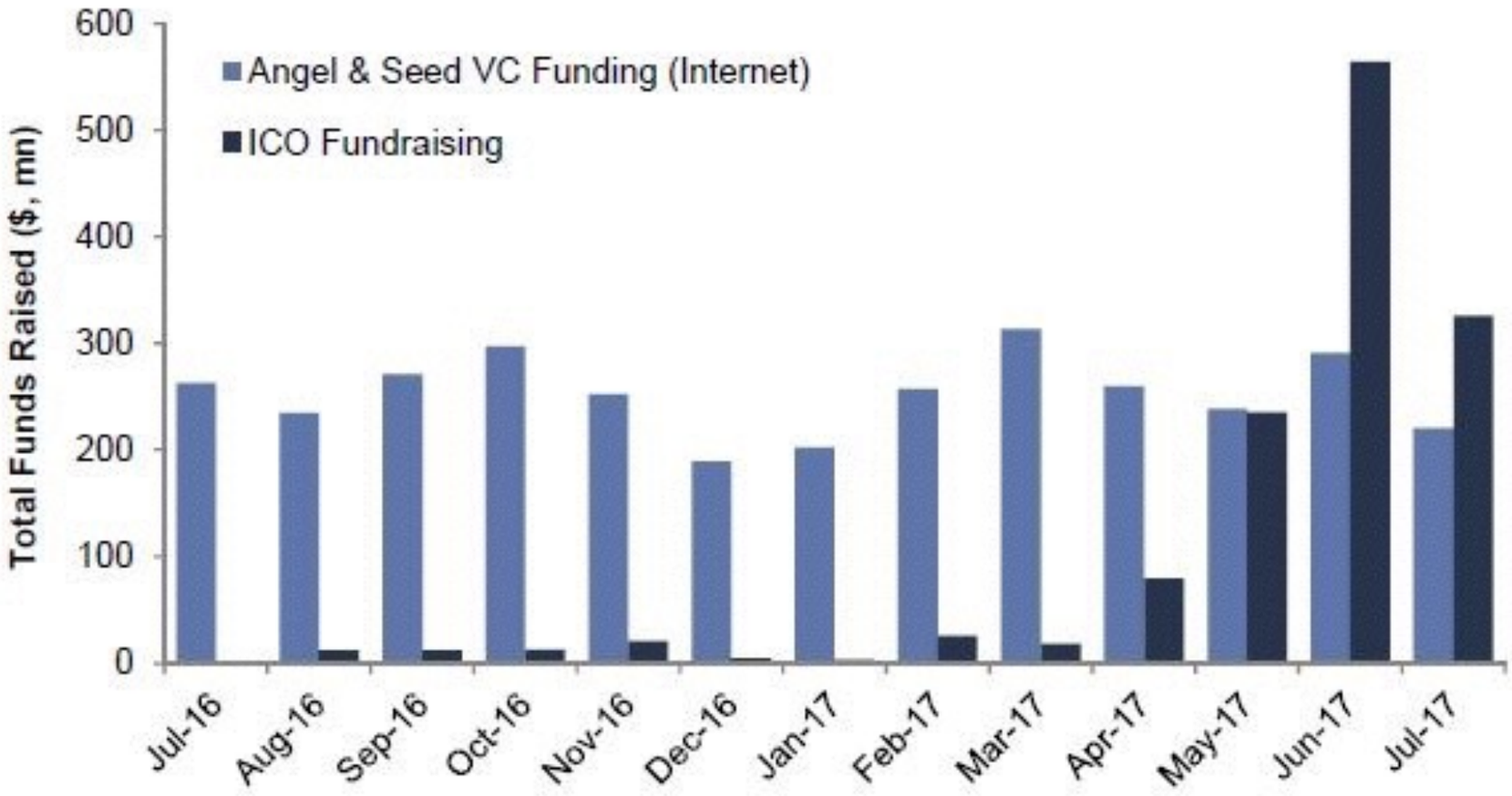
https://beta.theglobeandmail.com/report-on-business/kik-messenger-app-to-launch-its-own-bitcoin-like-digital-currency/article35108358/

CROWDFUNDING VIA TOKEN SALE

VERY EARLY, BUT SIGNS THAT IT COULD BE INTERESTING

Exhibit 8: The pace of ICO fundraising has now surpassed Angel & Seed stage Internet VC funding globally

Total Funds Raised by month ($, millions)



Note: ICO fundraising as of July 18th, 2017, per Coin Schedule. Angel & Seed VC funding data as of July 31st, 2017 and does not include "crowdfunding" rounds.

Source: CoinSchedule, CB Insights, Goldman Sachs Global Investment Research.

FREQUENTLY ASKED QUESTIONS (FAQ):

“

AREN’T THESE ILLEGAL?

— *Naysayers*

”

Exhibit: SEC Report (US Regulator)
https://www.sec.gov/oiea/investor-alerts-and-bulletins/ib_coinofferings

FREQUENTLY ASKED QUESTIONS (FAQ):

“

WHAT ABOUT SCAMS?

— *Naysayers*

”

Exhibit: Jucero https://www.youtube.com/watch?v=5lutHF5HhVA

WHAT DOES BLOCKCHAIN ENABLE?

A USER OR APPLICATION PERSPECTIVE

- Enables trust in trust-less environments
- Transfer value(!) within a peer-to-peer network of untrusted participants => “Internet of Value”
  - cryptocurrencies (ex. Bitcoin - among thousands of others)
- “Smart Contract” a computer program executed on a blockchain
- “Decentralized” Apps
- New classes of asset can be tokenized
- A useful buzzword like “the cloud”

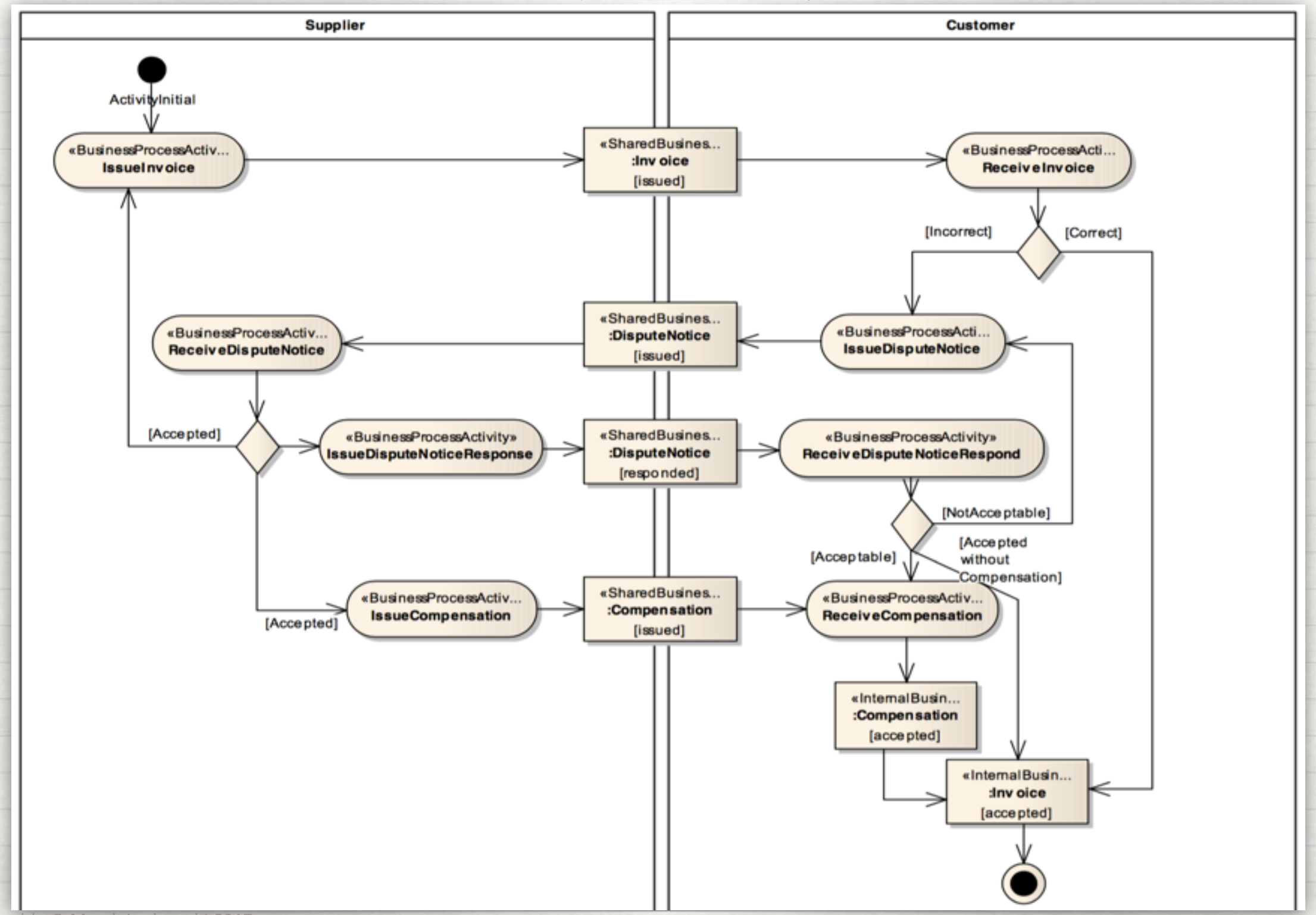
EXAMPLE

WORK WITH UNITED NATIONS (CEFACT)

- Exploratory work.
- Blockchain Revolution vs. Evolution.
- Full impact of blockchain may take 10-15 years to really be understood.
- How can blockchain augment existing business processes?
- Focus on core blockchain value proposition of shared, immutable version of the truth.

BUSINESS REQUIREMENTS SPECIFICATION (BRS)

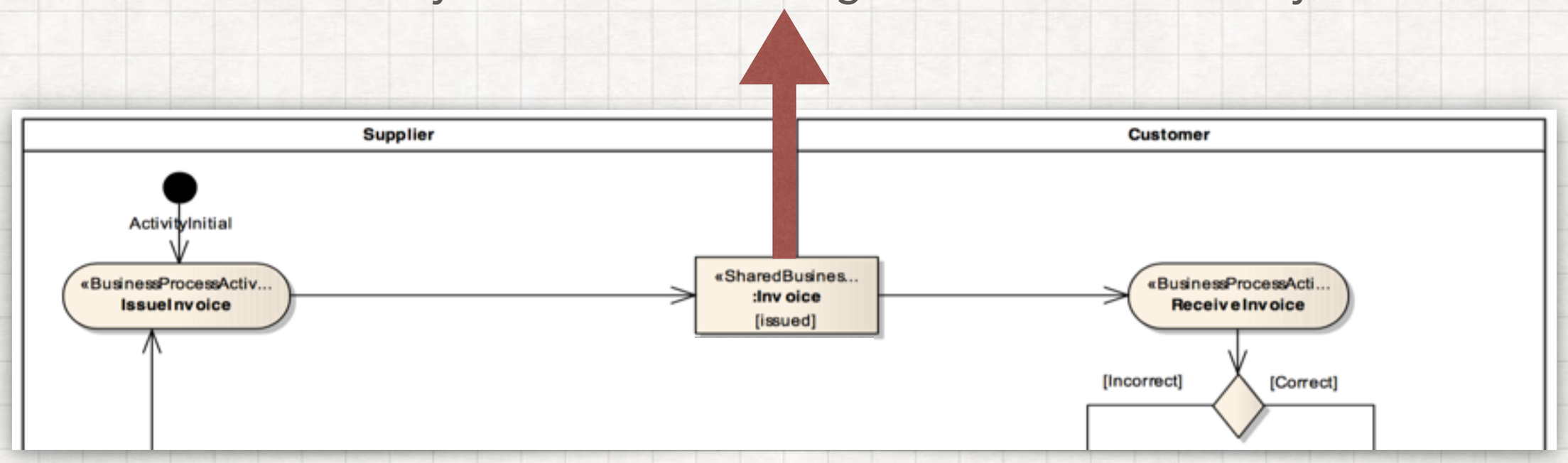
CROSS-INDUSTRY INVOICE



CROSS-INDUSTRY INVOICE (BRS)

SCENARIO 1: BLOCKCHAIN AS "NOTARY" SERVICE

Invoice data written to blockchain, signed by Supplier.
Potentially unrelated to message or document delivery.

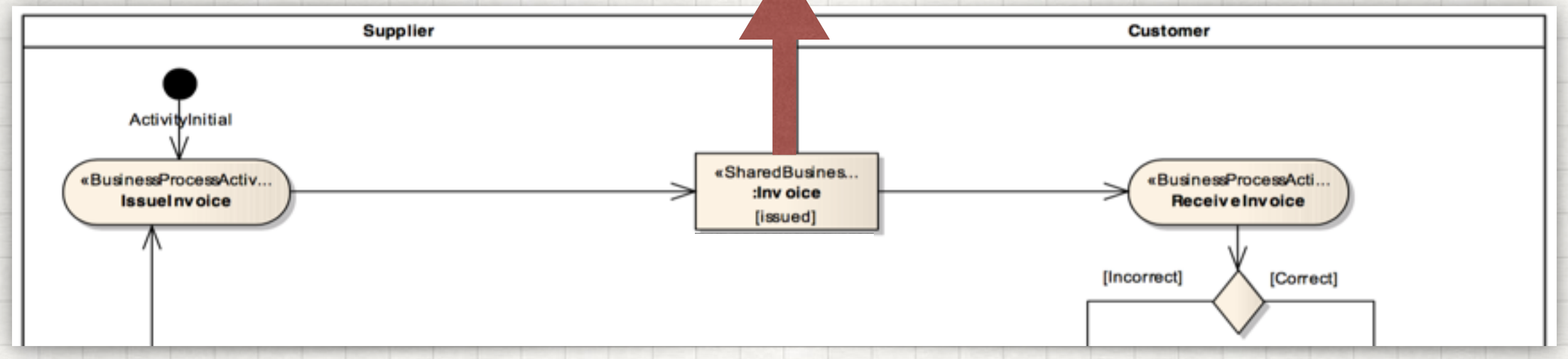


Transaction (Document) is stored as plain text, hash, or encrypted text written onto the blockchain.
Actor (supplier/customer) electronic signatures authenticate the transaction.
Provides Proof of existence of some document at some time.
Shared state or version of the truth.
e.g. Factom, OpenTimeStamps

CROSS-INDUSTRY INVOICE (BRS)

SCENARIO 1.5: BLOCKCHAIN AS "NOTARY" SERVICE, PAIRED WITH MESSAGE DELIVERY

Invoice data written to blockchain, signed by Supplier.
Encrypted document is retrievable by its hash.
Blockchain is closely tied to message delivery.



Transaction message hash is stored on the blockchain.
Actor (supplier/customer) electronic signatures authenticate the transaction.
Provides Proof of existence of some document at some time.
Shared state or version of the truth.
"Content Addressable Storage" (ex. IPFS, StorJ) to deliver the document/message.

ANATOMY OF A SMART CONTRACT

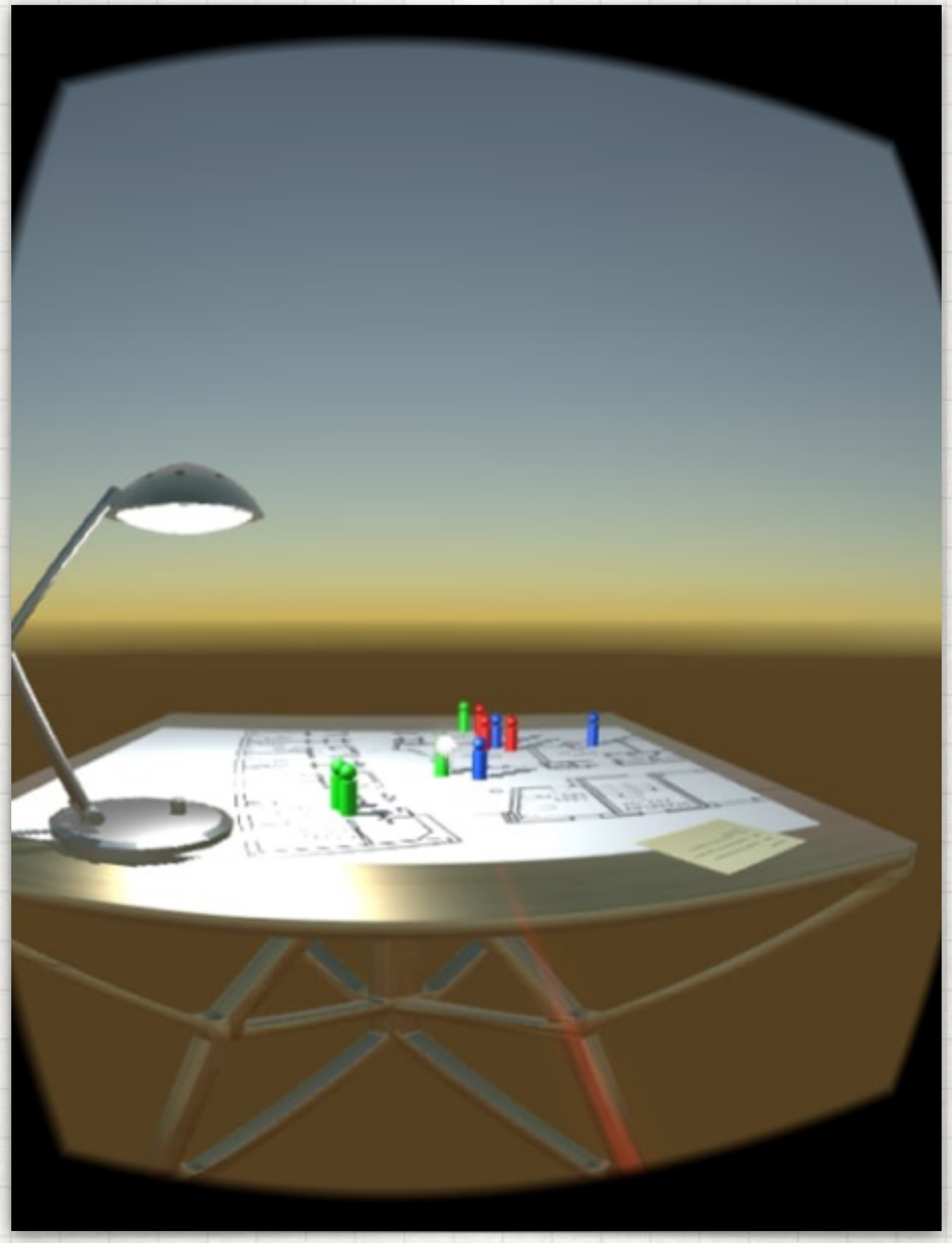
A LIMITED TRIAL IMPLEMENTATION UNDER DEVELOPMENT

- Ethereum/Solidity
- This code initializes the contract with defined customer and supplier roles.
- Business Process logic encoded as a Smart Contract (state transitions)
- Messages are signed and stored for auditability.
- Design permits separate contracts for customer and supplier that encode decision logic.

```
16 //Finite State Machine implements Business Process Activity
17 //Fig. 5-2 BRS Cross Industry - Supply Chain v2.00.05
18 contract CrossIndustryInvoice is StateTransitions{
19     //data
20     uint messageCount; //track messages or documents sent back and forth
21     string[] messages; //plain text or hash
22     address supplier; //could be human or
23     address customer; //Smart Contract!
24
25     function CrossIndustryInvoice(address supp, address cust){
26         supplier = supp;
27         customer = cust;
28         messageCount = 0;
29         //set up state mapping and initial state
30         setInitialState();
31     }
32
33     //handle supplier issuing invoice
34     function issueInvoice(string message){
35         require(msg.sender == supplier);
36         require(state == getState("Initial State"));
37         messageCount = messages.push(message);
38         state = getState("Invoice Issued");
39         notify(customer);
40     }
41     //handle customer accepting invoice
42     function acceptInvoice(string message){
43         require(msg.sender == customer);
44         require(state == getState("Invoice Issued"));
45         messageCount = messages.push(message);
46         state = getState("Invoice Accepted");
47         notify(supplier);
48     }
49     //handle customer disputing invoice
50     function issueDisputeNotice(string message){
51         require(msg.sender == customer);
52         require(state == getState("Invoice Issued"));
53         messageCount = messages.push(message);
54         state = getState("Invoice Disputed");
55         notify(supplier);
56     }
```

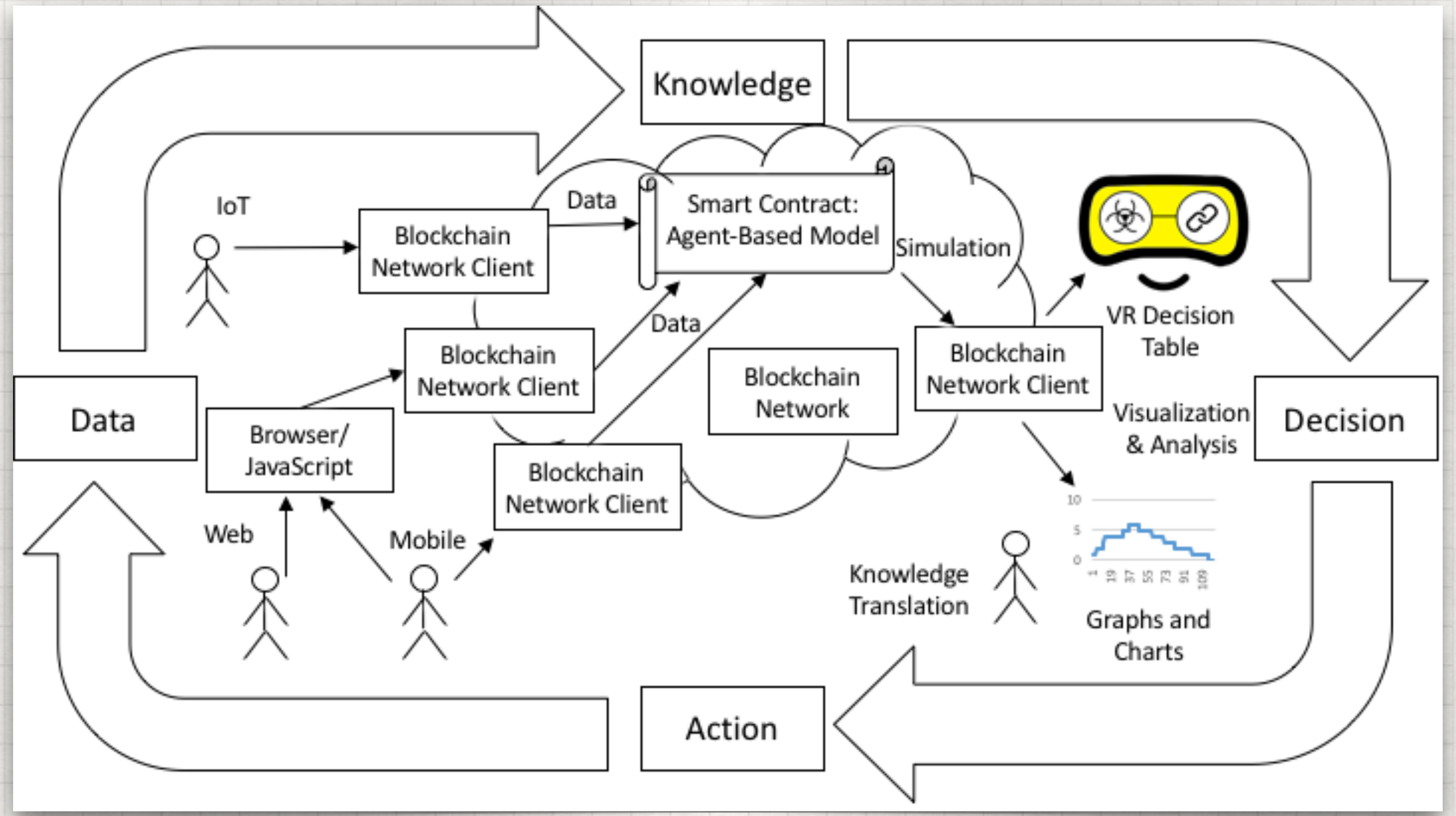
ANOTHER EXAMPLE:

DECISION SUPPORT



SYSTEM DIAGRAM

PARTICIPATORY DECISION SUPPORT USING BLOCKCHAIN



OTHER APPLICATIONS

YOU CAN THINK OF MORE!

- Supply chain, logistics, traceability (https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2828369)
- Financial Technology (Fintech)
- Real estate, land titles, structuring deals with Smart Contracts, fractional ownership, crowdfunding
- Health - data records management
- Accounting
- Insurance
- Energy / Carbon Credits
- Can Blockchain replace trusted third parties (middlemen) with a trust-less network

CRYPTOGRAPHIC PRINCIPLES

PUBLIC KEY CRYPTOGRAPHY

FOUNDATIONS FOR TRUST ON THE INTERNET

- Cryptography is founded in specialized math.
- Offline, a user can generate a public-private key-pair.
- Math and huge number space (http://directory.io/) guarantees that no two users will have the same key.
- Typically, good public keys are just that, as widely distributed as possible!
- Private keys need to be guarded. If these leak out, all is lost.
- These schemes (ex. RSA) underlie most secured transactions on the Internet (ex. web banking)

ENCRYPTION AND DECRYPTION

APPLICATIONS OF CRYPTOGRAPHY

- Ex. Alice wants to send a message to Bob
- Bob gives Alice (publishes) his public key
- Alice uses Bob's public key to encrypt the message
- Alice sends Bob the encrypted message
- Bob decrypts the message with his private key
- Nobody besides the holder of the public key (and Alice!) can read the message.

DIGITAL SIGNATURES

APPLICATIONS OF CRYPTOGRAPHY

- Digital signatures are used to verify the authenticity of the message i.e. that it came from the claimed sender.
- Ex. Alice wants to make sure that a message came from Bob:
- Bob gives Alice his public key
- Bob signs his message with his private key
- Bob sends Alice the message and signature
- Alice verifies the signature with Bob’s public key

HASHING FUNCTIONS

APPLICATIONS OF CRYPTOGRAPHY

- A hashing function is a so-called one-way function that transforms an input message into a message digest (called the hash)
- It's very difficult to reproduce the input that resulted in a particular hash
- A small change in the input message will result in an unpredictable large change in the output hash
- It's very unlikely to find two input messages with the same hash
- You can prove you have a message or it existed at a certain time
- Ex. MD5

PROOF OF WORK

BRINGING IT ALL TOGETHER

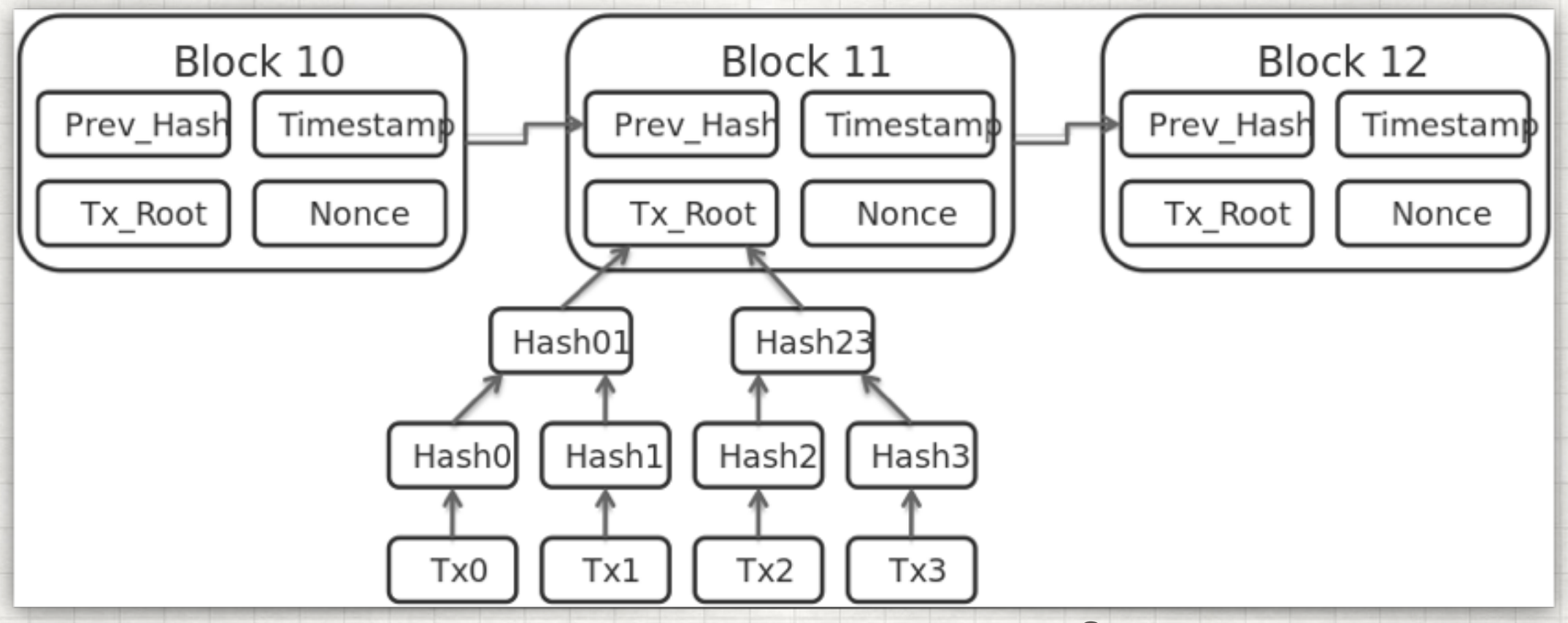
- Recall that hashes are essentially unpredictable with respect to the input data.
- The protocol adjusts the number of leading zeroes required in the hash of the next block to keep the blocks spaced out in time regardless of the number and computational power of the miners.
- Miners add a random padding (nonce) to the data in order to attempt win the “lottery”.
- You can prove that on average, a particular amount of work, as counted by hash function attempts to beat the target.
- In Bitcoin and Ethereum, miners (those maintaining the ledger, by hashing blocks) are rewarded with newly minted tokens and transaction fees awarded to the corresponding public key that signed the block.

BITCOIN BLOCKS IN THE BLOCKCHAIN

(ETHEREUM IS CONCEPTUALLY SIMILAR)

Miners organize all transactions received since the last block in
a data structure called a Merkle Tree for inclusion in the current block.

Merkle Trees make it efficient to check whether a transaction hash is within the block.



Source:

By Matthäus Wander - Own work, CC BY-SA 3.0, https://commons.wikimedia.org/w/index.php?curid=26816920

HISTORY OF BLOCKCHAIN

BITCOIN CIRCA 2009

NOTE: NOT THE FIRST ATTEMPT AT DIGITAL CURRENCY

- Satoshi Nakamoto (pseudonym) launched Bitcoin January 03, 2009
- Embedded in the first block:
  - “The Times 03/Jan/2009 Chancellor on brink of second bailout for banks.”
- https://bitcoin.org/bitcoin.pdf “A Peer-to-Peer Electronic Cash System”
- Value grew with energy expended to secure the network (more on this later)
- Today: ~100B (yes, Billions of USD) “Market Cap” comparable to many large corporations, smaller fiat currencies (!)

MOTIVATION FOR ETHEREUM

QUEST FOR A BETTER MOUSETRAP

- Bitcoin is specialized and very good at securing the ownership of Bitcoins.
- However, it's possible that more complex interactions are beneficial.
- For example, encoding business logic into blockchain transactions, so-called Smart Contracts
- Ethereum's design goal was to create a one-world computer, and make it easier to write Smart Contracts by virtue of a more fully featured (and consequently more complex) execution model.
- Ethereum's Virtual Machine (EVM) and scripting languages are Turing Complete.
- Ethereum's mining algorithm uses a random Directed Acyclic Graph to make it difficult to create ASIC miners as are primarily used in Bitcoin today.

ETHEREUM HISTORY

SHOUT OUT TO TORONTO

- Originally described by Vitalik Buterin (Canadian-Russian) - former Bitcoin developer - in 2013
- https://github.com/ethereum/wiki/wiki/White-Paper A Next-Generation Smart Contract and Decentralized Application Platform
- Ethereum was crowdfunded using Bitcoin (the original ICO). Funders got the original allocation of Ether tokens (ETH) in the genesis block
- The above white paper also interestingly proposes a sort of decentralized hedge fund called a Decentralized Autonomous Organization (DAO) whose rules are encoded by Smart Contracts and executed/enforced by the Ethereum Blockchain.

THE DAO

NEVER FORGET

- Soon after the launch of Ethereum slock.it (who was making digital locks on the blockchain) launched The DAO.
- It raised a record crowdfunding amount of $150,000,000
- But wait…
- There was a bug.
- An individual was able to exploit the bug to potentially allow them to take for themselves $50,000,000 of the contract's value.
- The Securities and Exchange Commission (USA) decided the DAO was an unregistered security but will not pursue the case (ICO's on notice!)

THE DAO FORK

AND A TAXONOMY OF BLOCKCHAIN FORKS

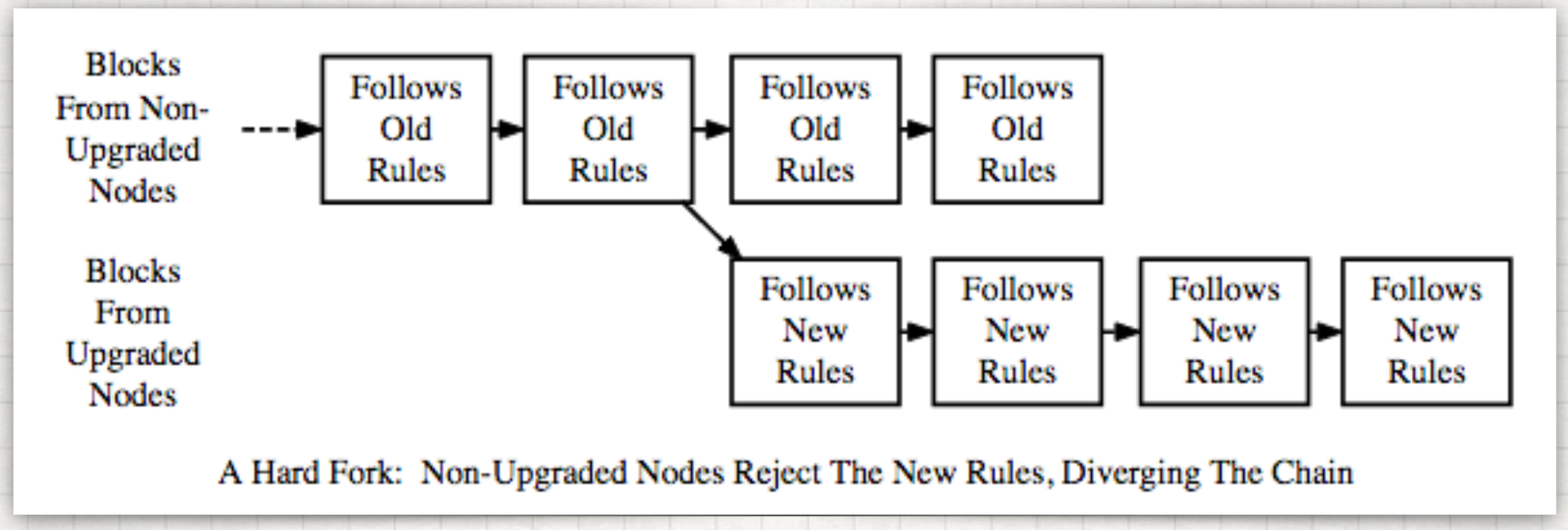
- It was decided that the “immutable ledger” would be overwritten to undo the theft and generally crappy Smart Contract
- You can see the wreckage here: https://etherscan.io/address/0xbb9bc244d798123fde783fcc1c72d3bb8c189413
- A “Hard Fork” occurs when miners update the protocol to reject previously valid blocks.
- (A “Soft Fork” loosens rules to permit previously invalid blocks)
- This contentious Hard Fork led to a “Chain Split” where some miners and community members did not agree to change the immutable ledger.
- This led to the recognition of Ethereum Classic (ETC) that exists to this day that continues to mine (produce new blocks on top of) the unaltered chain.

CURRENT EVENTS

A RAPIDLY EVOLVING TECHNOLOGY AND ECOSYSTEM

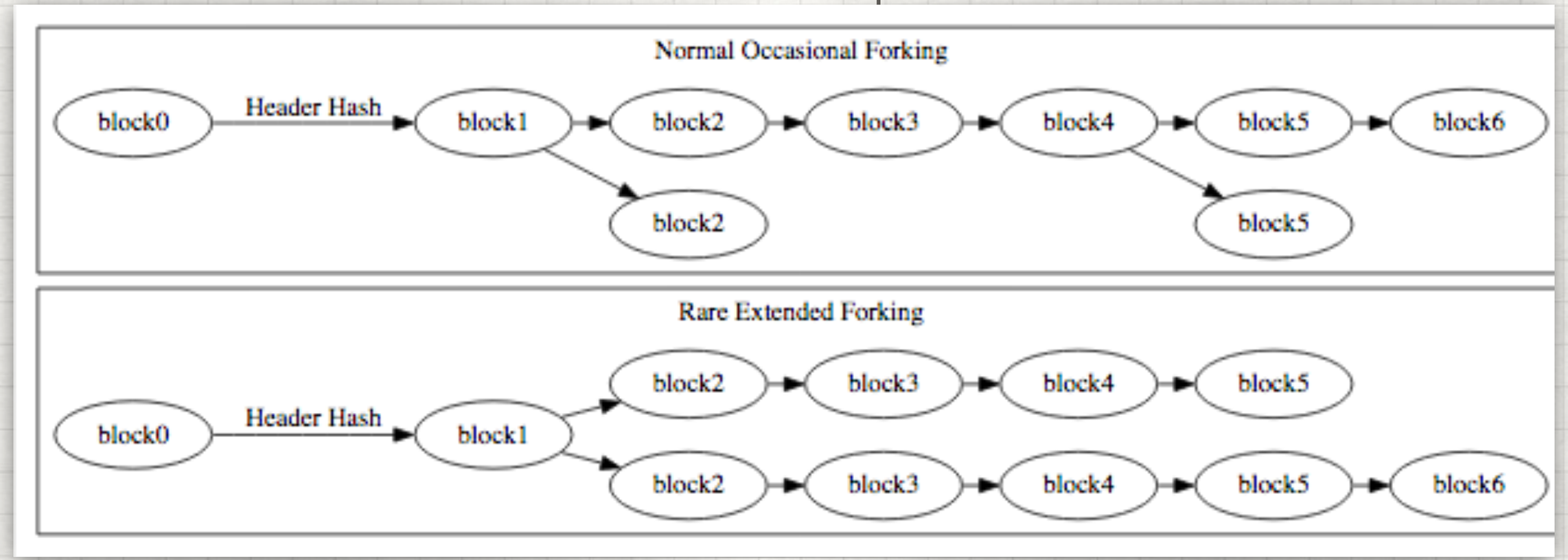
- It's the intention of creator Vitalik Buterin to move the system away from Proof-of-Work to Proof-of-Stake.
- Proof-of-Stake uses accumulated tokens instead of CPU cycles to "vote" for which miner produces the next block.
- Proof of Stake has ***not*** been demonstrated in production "in the wild".
- Part of this plan is to reduce the mining rewards and increasing the block times in the so-called Ethereum "Ice Age" that was implemented in the Byzantium Hard Fork in October, 2017.
- This fork was not contentious and did not result in the creation of another network chain split off.

MORE ON FORKS



A Hard Fork: Non-Upgraded Nodes Reject The New Rules, Diverging The Chain

source: Investopedia



source: bitcoin.org

WHY FORKS OCCUR

BEYOND INTENTIONAL FORKS

- The current state of the Blockchain is defined as the chain with the most work, usually the longest chain.
- Two miners could simultaneously solve the next block and then other miners would start to mine either chain randomly.
- Eventually one will win out as having the most work and is a normal part of the blockchain operation.
- This implies that the guarantees of blockchains are actually probabilistic!
- “Eventual Consensus” (Eventual Consistency in database terminology).
- It’s not uncommon to wait for several blocks on top of the block containing an important transaction (e.g. 6 in Bitcoin) to prevent double-spending of Bitcoins.

ETHEREUM

BASICS

SOME DETAILS ABOUT ETHEREUM

(BEFORE WE GET OUR HANDS DIRTY)

- All users have addresses (public-private key-pairs), so called “external” accounts.
- Ethereum also has Smart Contracts, which are programs deployed and executed by the Ethereum Virtual Machine (EVM) on the Ethereum blockchain.
- Smart Contracts also have addresses.
- Both kinds of accounts can have a balance of ETH
- However, all transactions have to be initiated by an External account.
- Transactions are paid for by the user, the fees are called “Gas”

MORE ON ETHEREUM

DETAILS.. DETAILS

- All operations have a particular gas cost expressed in various denominations of Ether (http://ethdocs.org/en/latest/ether.html)
- When we run out of gas, any changes resulting from the execution of the transaction (including state changes to Smart Contracts) are rolled back
- Exceptions occur when there's an unrecoverable error during the execution of a contract. It could be a bug or a violation of a condition such that only the owner call a certain function
- Exceptions -> state is rolled back (Ex. running out of gas)
- Uncle blocks - the ethereum protocol rewards finding small forks and reporting blocks which recently forked from the longest chain.

OVERVIEW OF THE ECOSYSTEM

WE'LL GET OUR HANDS DIRTY SOON ENOUGH

- Public network (Main network)
- Exposed and browsable on the web.
- https://ethstats.net/ (network health and status)
- http://etherscan.io/ (view transaction and block data)
- https://etherscan.io/address/0xbb9bc244d798123fde783fcc1c72d3bb8c189413
- Private or Permissioned Blockchains are possible for enterprise users
- Enterprise Ethereum Alliance https://entethalliance.org/

OVERVIEW OF THE ECOSYSTEM

- Ethereum Foundation - promote and support the protocol
- https://www.ethereum.org/foundation
- “ConsenSys is a venture production studio building decentralized applications and various developer and end-user tools for blockchain ecosystems, primarily focused on Ethereum”
- https://consensys.net/
- Both funded through initial crowdfunding effort.

MULTIPLE IMPLEMENTATIONS

ROBUSTNESS THROUGH DIVERSITY

- The node and mining software has multiple implementations of the protocol. Geth, Parity, etc.
- Multiple languages for Smart Contract Development Solidity, ~~Serpent~~
- Mist Browser
- MetaMask - Web Browser plugin
- Solc - solidity compiler (inconvenient!)
- Remix IDE - web-based IDE with built-in compiler (sweet!)
- https://remix.ethereum.org