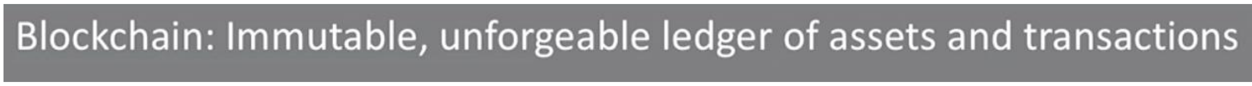
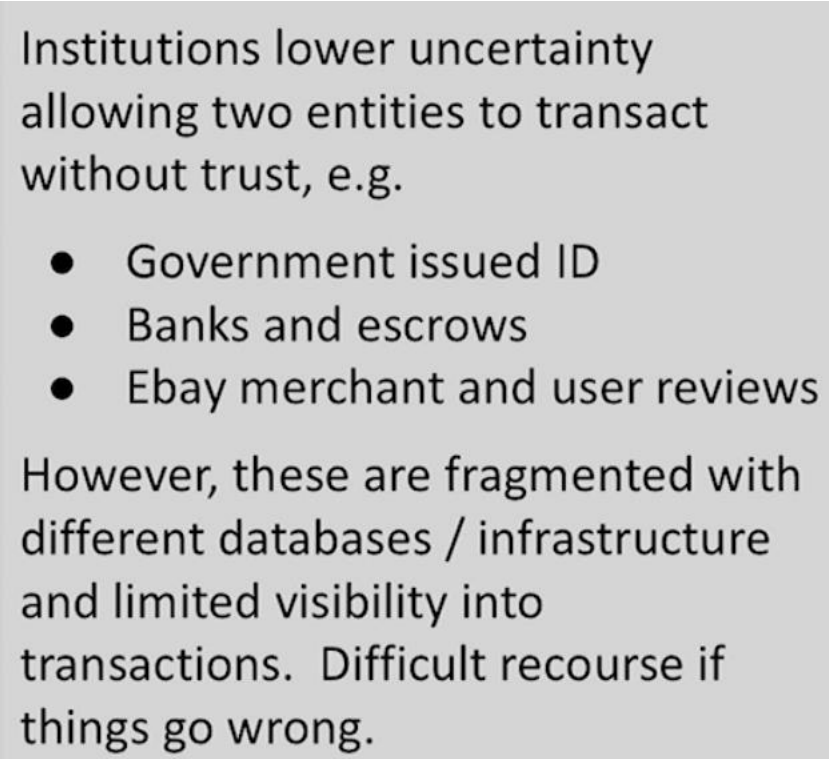
Blockchain Applications and Smart Contracts

1. Introduction to Blockchains and Smart Contracts

- **Explain the history of blockchain technology:**



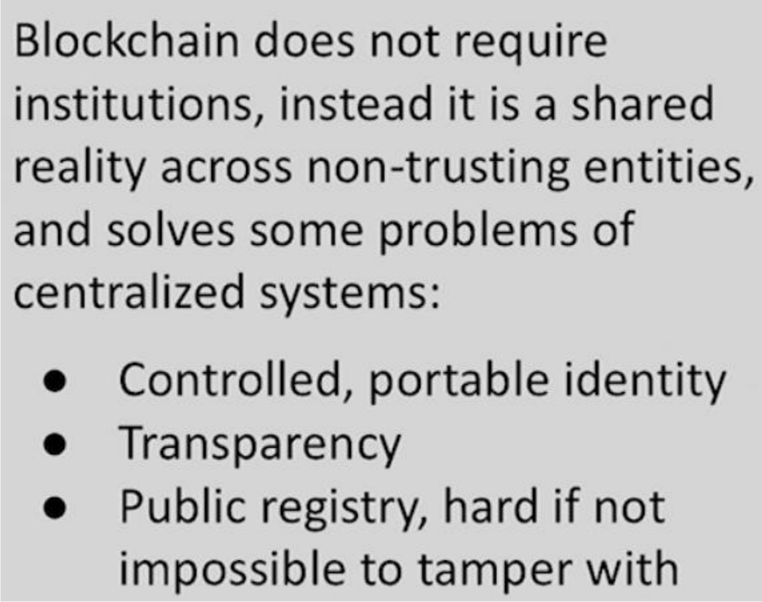
Blockchain: Immutable, unforgeable ledger of assets and transactions



Institutions lower uncertainty allowing two entities to transact without trust, e.g.

- Government issued ID
- Banks and escrows
- Ebay merchant and user reviews

However, these are fragmented with different databases / infrastructure and limited visibility into transactions. Difficult recourse if things go wrong.

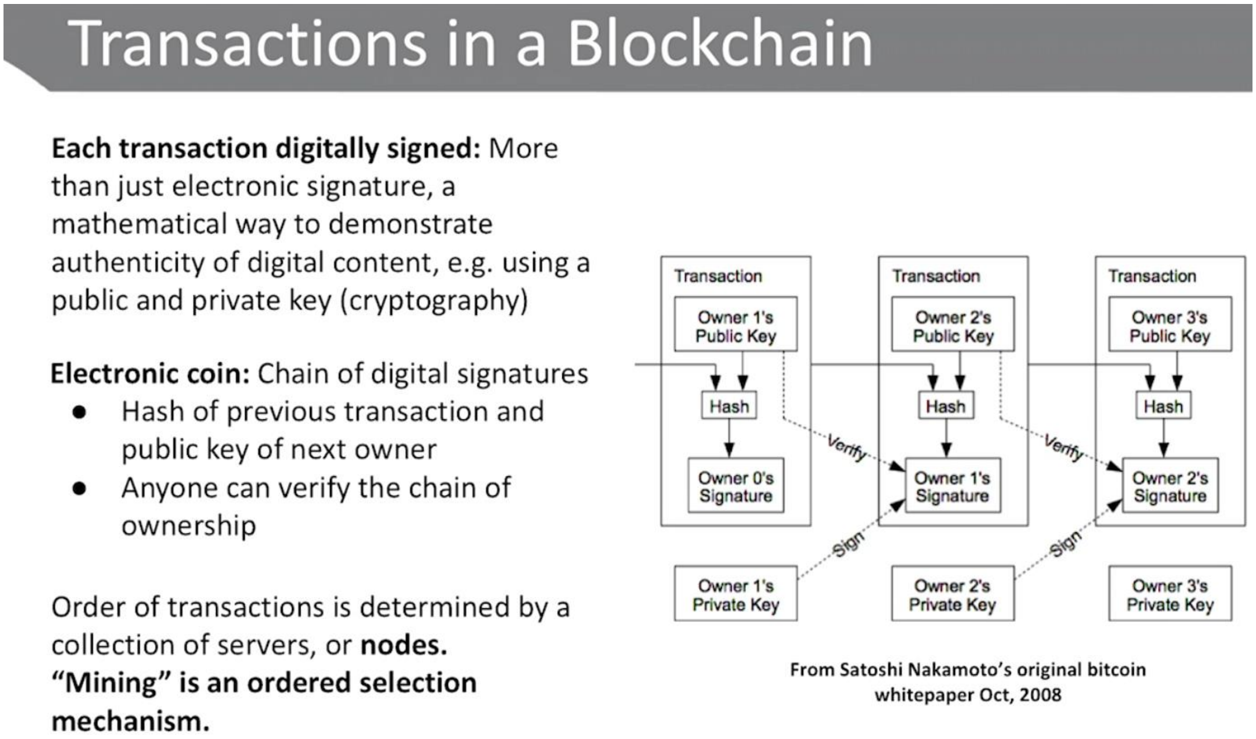


Blockchain does not require institutions, instead it is a shared reality across non-trusting entities, and solves some problems of centralized systems:

- Controlled, portable identity
- Transparency
- Public registry, hard if not impossible to tamper with

Blockchain Applications and Smart Contracts

> I've been working on a new electronic cash system that's fully peer-to-peer, with no trusted third party.



Transactions in a Blockchain

Each transaction digitally signed: More than just electronic signature, a mathematical way to demonstrate authenticity of digital content, e.g. using a public and private key (cryptography)

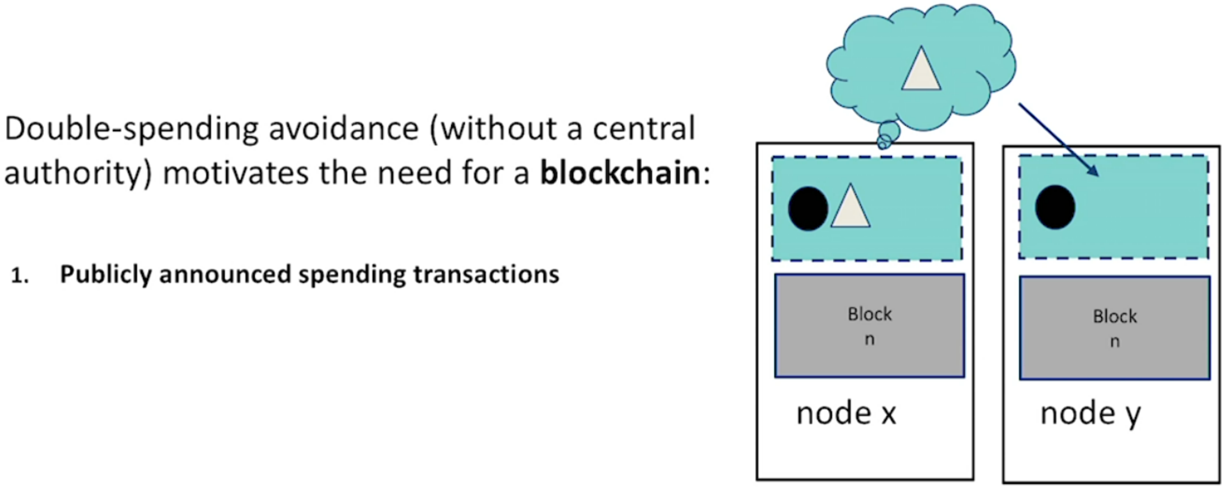
Electronic coin: Chain of digital signatures

- Hash of previous transaction and public key of next owner
- Anyone can verify the chain of ownership

Order of transactions is determined by a collection of servers, or **nodes.** **"Mining" is an ordered selection mechanism.**

From Satoshi Nakamoto's original bitcoin whitepaper Oct, 2008

- **Understand the consequences of double-spending avoidance**

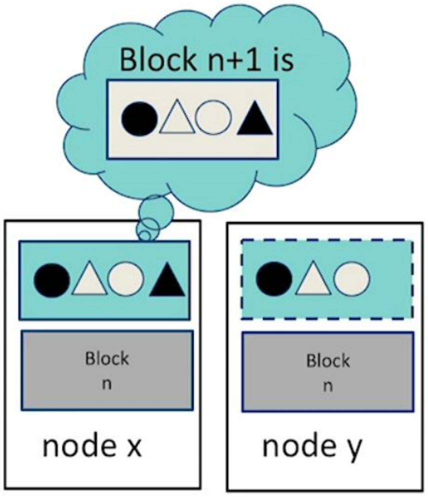


Double-spending avoidance (without a central authority) motivates the need for a **blockchain**:

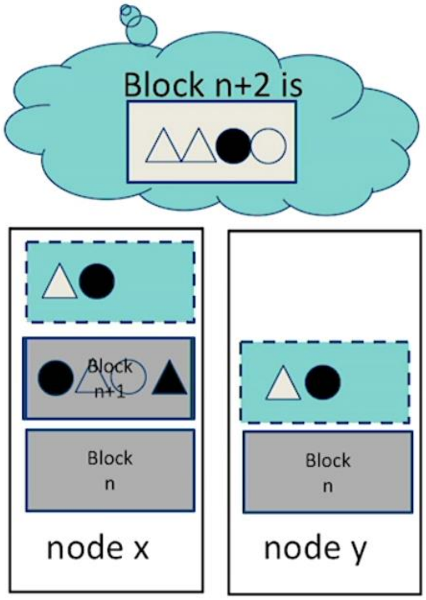
1. **Publicly announced spending transactions**

Blockchain Applications and Smart Contracts

2. Each node keeps track of a chain of blocks of transactions (mining)
   a. Competes for completed block of transactions (proof of work)
   b. Broadcasts each completed block to all other nodes
   c. Accepts broadcast block only if all transactions are not already spent
   d. Starts building the next block based on this



3. If any discrepancies exist between nodes, the longest chain wins, and invalid blocks are reverted, abandoning the later duplicate transactions



Overall, the underlying mechanism does not need to be understood, but it provides a motivation to learn many aspects of the technology.

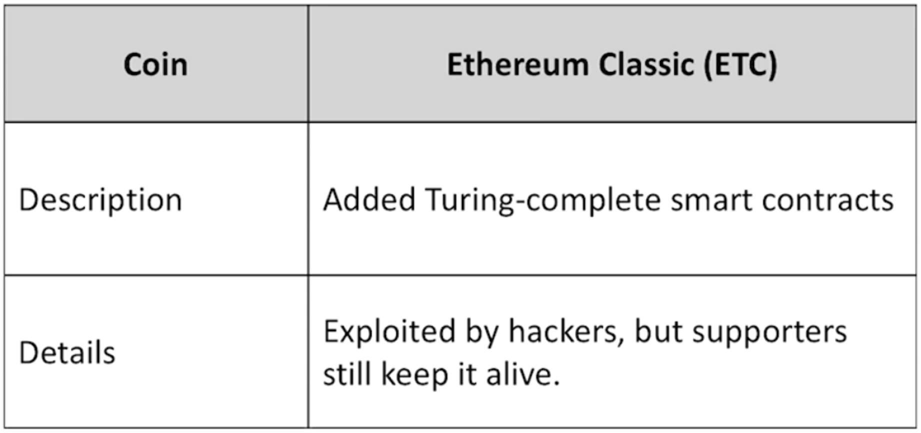
A conventional blockchain is:

Public…by default: A decentralized, peer-to-peer ledger without trust. But private blockchains can be set up in a similar manner.

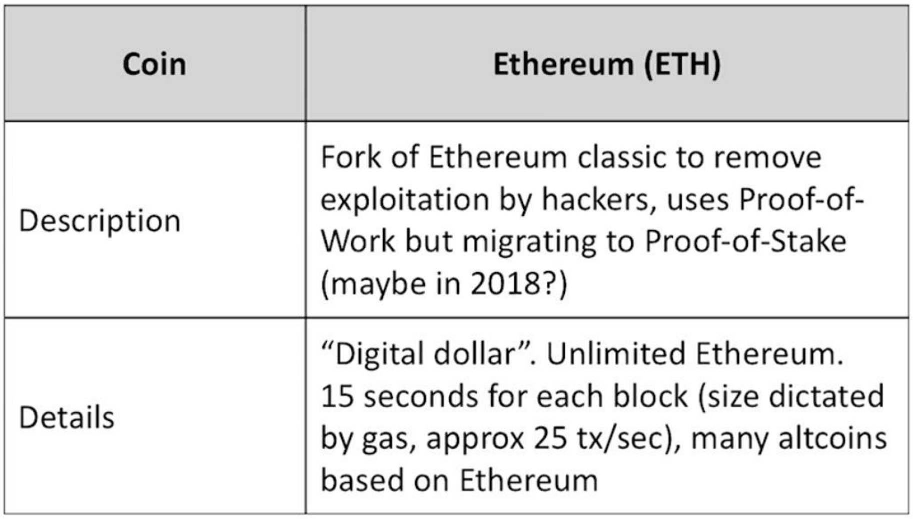
Immutable…over time: Consensus is built through mining. Need 6 confirmations to be 99.9% sure of the transaction, so this takes an hour for Bitcoin and 1.5 minutes for Ethereum.

Blockchain Applications and Smart Contracts

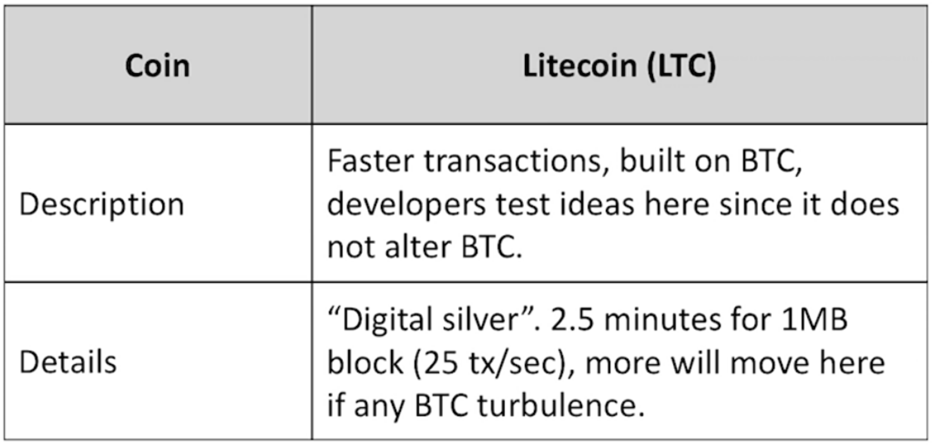
- **Appreciate the objective of different blockchains:**



Coin	Ethereum Classic (ETC)
Description	Added Turing-complete smart contracts
Details	Exploited by hackers, but supporters still keep it alive.

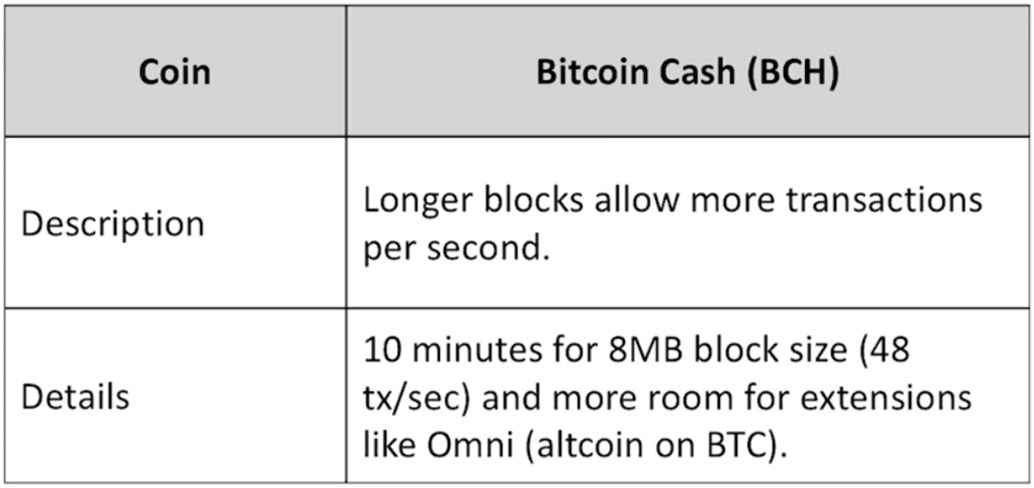


Coin	Ethereum (ETH)
Description	Fork of Ethereum classic to remove exploitation by hackers, uses Proof-of-Work but migrating to Proof-of-Stake (maybe in 2018?)
Details	“Digital dollar”. Unlimited Ethereum. 15 seconds for each block (size dictated by gas, approx 25 tx/sec), many altcoins based on Ethereum

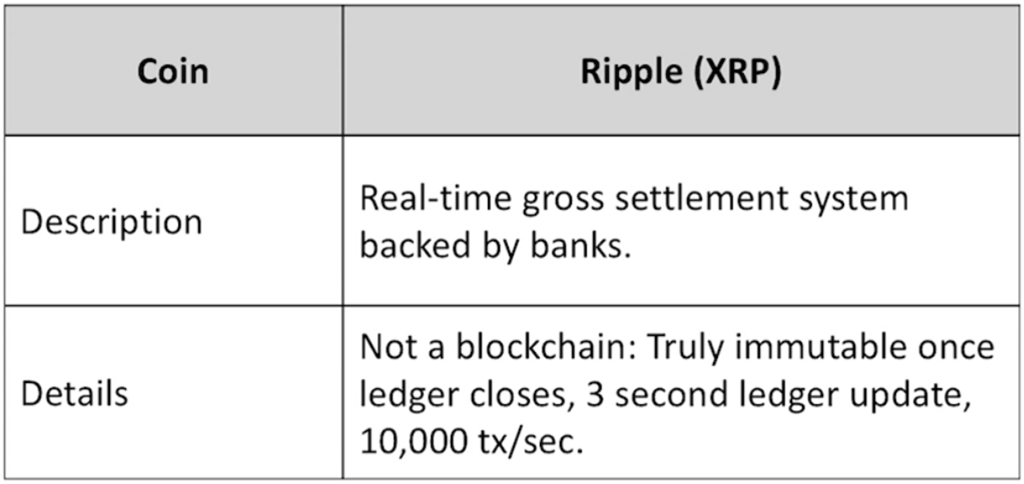


Coin	Litecoin (LTC)
Description	Faster transactions, built on BTC, developers test ideas here since it does not alter BTC.
Details	“Digital silver”. 2.5 minutes for 1MB block (25 tx/sec), more will move here if any BTC turbulence.

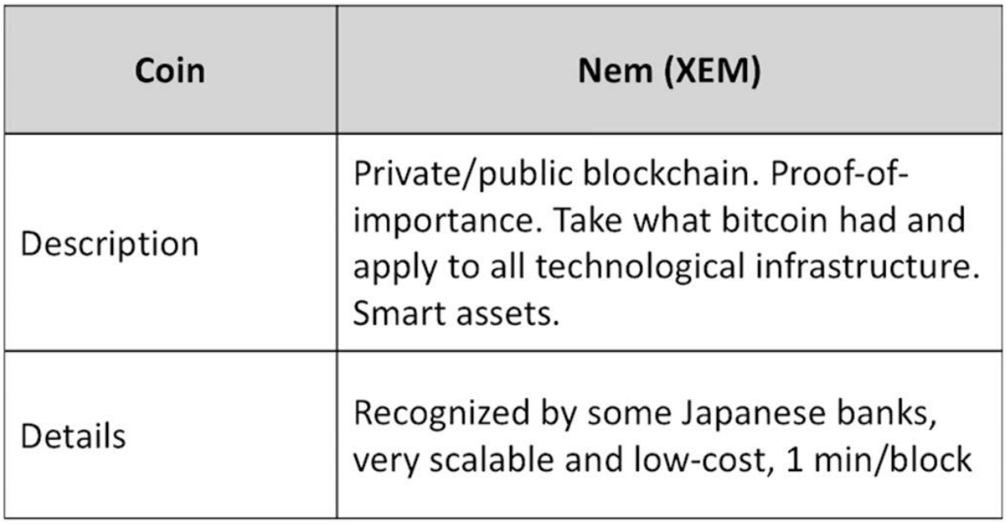
Blockchain Applications and Smart Contracts



Coin	Bitcoin Cash (BCH)
Description	Longer blocks allow more transactions per second.
Details	10 minutes for 8MB block size (48 tx/sec) and more room for extensions like Omni (altcoin on BTC).



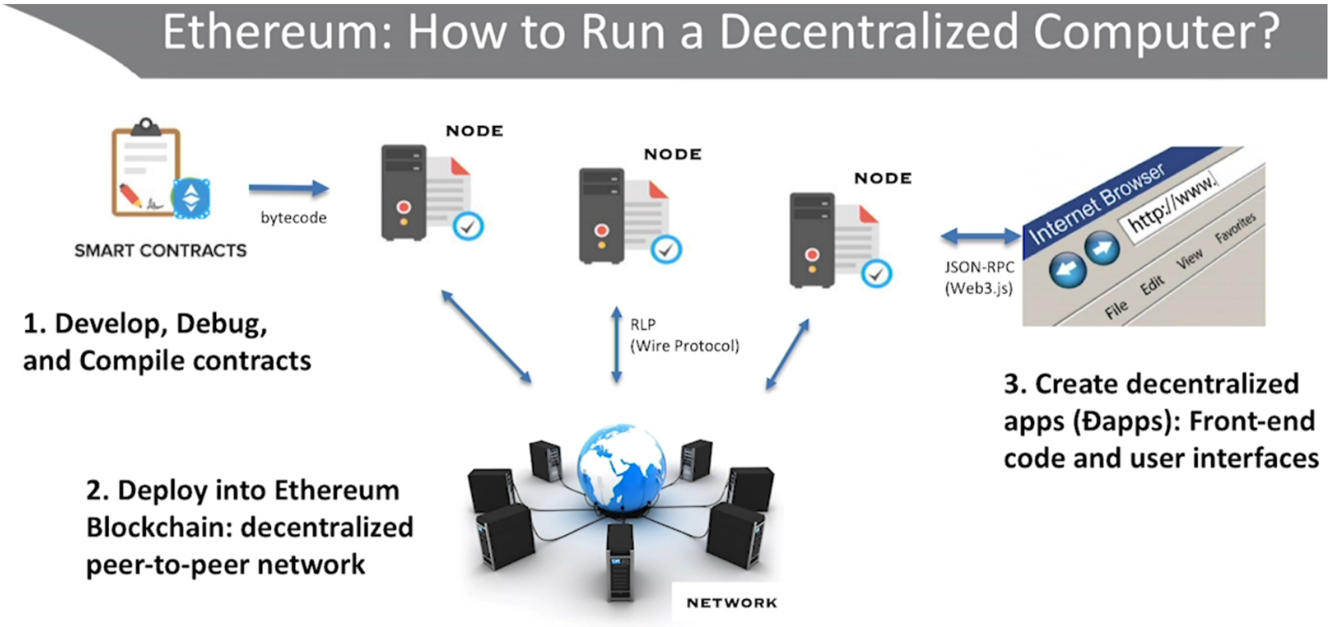
Coin	Ripple (XRP)
Description	Real-time gross settlement system backed by banks.
Details	Not a blockchain: Truly immutable once ledger closes, 3 second ledger update, 10,000 tx/sec.



Coin	Nem (XEM)
Description	Private/public blockchain. Proof-of-importance. Take what bitcoin had and apply to all technological infrastructure. Smart assets.
Details	Recognized by some Japanese banks, very scalable and low-cost, 1 min/block

Blockchain Applications and Smart Contracts

- **Add smart contracts to blockchains**



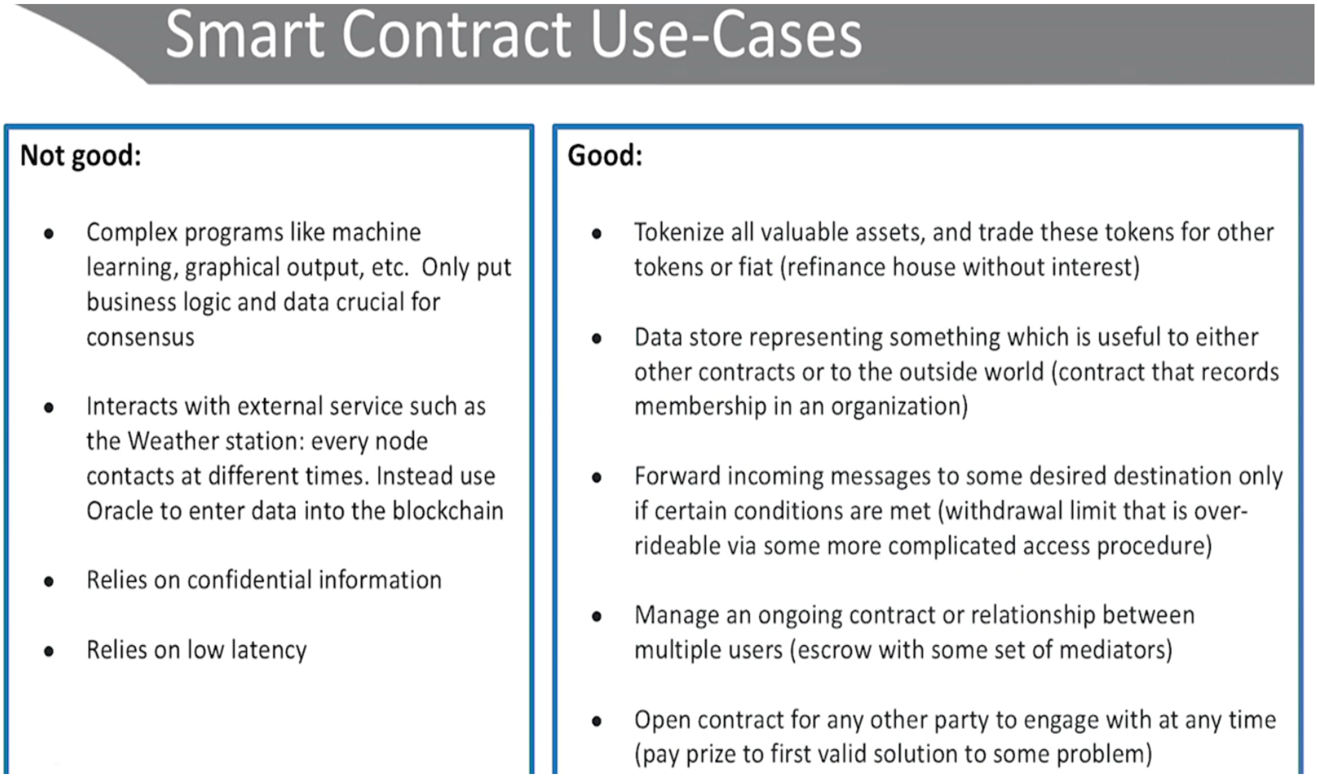
Ethereum: How to Run a Decentralized Computer?

1. Develop, Debug, and Compile contracts

2. Deploy into Ethereum Blockchain: decentralized peer-to-peer network

3. Create decentralized apps (Ðapps): Front-end code and user interfaces

- **Determine relevant smart contract use-cases**



Smart Contract Use-Cases

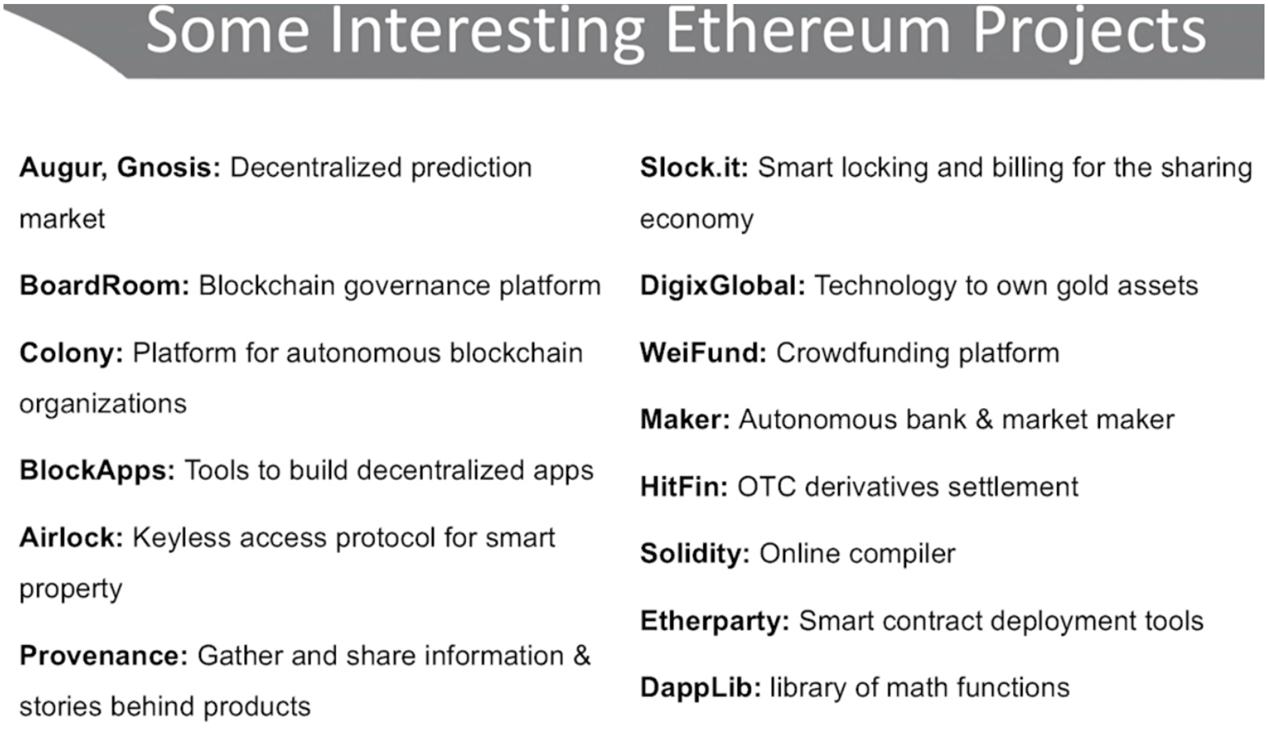
Not good:

- Complex programs like machine learning, graphical output, etc. Only put business logic and data crucial for consensus
- Interacts with external service such as the Weather station: every node contacts at different times. Instead use Oracle to enter data into the blockchain
- Relies on confidential information
- Relies on low latency

Good:

- Tokenize all valuable assets, and trade these tokens for other tokens or fiat (refinance house without interest)
- Data store representing something which is useful to either other contracts or to the outside world (contract that records membership in an organization)
- Forward incoming messages to some desired destination only if certain conditions are met (withdrawal limit that is over-rideable via some more complicated access procedure)
- Manage an ongoing contract or relationship between multiple users (escrow with some set of mediators)
- Open contract for any other party to engage with at any time (pay prize to first valid solution to some problem)

Blockchain Applications and Smart Contracts



Some Interesting Ethereum Projects

Augur, Gnosis: Decentralized prediction market

BoardRoom: Blockchain governance platform

Colony: Platform for autonomous blockchain organizations

BlockApps: Tools to build decentralized apps

Airlock: Keyless access protocol for smart property

Provenance: Gather and share information & stories behind products

Slock.it: Smart locking and billing for the sharing economy

DigixGlobal: Technology to own gold assets

WeiFund: Crowdfunding platform

Maker: Autonomous bank & market maker

HitFin: OTC derivatives settlement

Solidity: Online compiler

Etherparty: Smart contract deployment tools

DappLib: library of math functions

2. **Ethereum: A Smart Contract Blockchain**
3. **Solidity: A Contract-Oriented Language**
4. **Testing, Debugging, and Deploying Smart Contracts**
5. **Smart Contracts Example: a Custom Token in Ethereum**